

Thèse de doctorat

Pour obtenir le grade de : Docteur de l'université de Paris 5 René Descartes

Discipline : Informatique

Présentée par : Rudolf Pareti

Indexation d'images par une loi puissance

Présentée le :

Alain Faure	Examineur
Jean-Marc Ogier	Rapporteur
Karl Tombre	Rapporteur
Georges Stamon	Examineur
Nicole Vincent	Directrice de thèse
Yue min Zhu	Examineur

REMERCIEMENTS

RÉSUMÉ

Avec l'évolution croissante du multimédia et des technologies, le besoin d'indexer les données est de plus en plus présent dans notre monde. Alors qu'avant les disques durs et autres supports numériques n'étaient que de simples espaces de stockage peu volumineux, ils sont devenus avec leur évolution des outils de sauvegarde de masse. L'image est une donnée multimédia incontournable et est la chose la plus présente sur les disques durs, non seulement des professionnels mais aussi des particuliers. Les besoins grandissant de la sauvegarde du patrimoine documentaire de l'humanité amènent les différents acteurs de cette sauvegarde à numériser en masse les vieux ouvrages de notre culture. C'est dans ce cadre que nous présentons ici une nouvelle méthode pour indexer efficacement des images numériques, non seulement de documents anciens mais aussi d'images naturelles. Notre méthode repose sur l'utilisation d'une loi puissance nommée loi de Zipf déjà utilisée dans d'autre domaine que l'indexation d'image.

TABLE DES MATIERES

1. Introduction	9
1.1 Contexte et motivations.....	9
1.2 Contributions et organisation du rapport	14
2 État de l'art de l'indexation d'images	15
2.1 Définitions	15
2.2 Indexation par les métadonnées	17
2.3 Indexation par le contenu.....	19
2.4 Approche structurelle	22
2.4.1 Segmentation par fenêtre	24
2.4.2 Segmentation par régions homogènes.....	25
2.4.3 Extraction d'objet par une fenêtre de corrélation	25
2.4.4 Rétroprojection de primitives	27
2.5 Approche Structurelle avec extraction des points d'intérêt.....	28
2.5.1 Détection de BLOB.....	29
2.5.2 Détection des coins	29
2.6 Approches statistiques.....	32
2.6.1 Les histogrammes	33
2.6.2 Les moments statistiques et histogrammes cumulés	37

2.6.3	Les champs de Markov	38
2.6.4	Les invariants	38
2.6.5	Bilan	39
3	Les lois puissance	41
3.1	Définition	42
3.2	Les principales lois puissances	45
3.2.1	La loi de Pareto	45
3.2.2	La loi de Benford	47
3.2.3	Fractales	48
3.2.4	La loi de Zipf	50
3.2.5	La loi de Zipf inverse	55
3.3	Application des lois puissances	57
4	Le codage des motifs	59
4.1	Le choix du motif	59
4.2	La quantification des couleurs	61
4.2.1	Les 9 classes	61
4.2.2	Les rangs généraux	62
4.2.3	Les K-Means	65
4.2.4	Contraste et gradient	66
5	Interprétation du graphe de Zipf	73
5.1	Les Rangs généraux	73
5.1.1	Les rangs généraux et la loi de Zipf	74
5.1.2	Les rangs généraux et la loi de Zipf inverse	81
5.2	Les classes	83
5.2.1	Les classes et la loi de Zipf	84
5.2.2	Les classes couleur	97

5.2.3	La quantification et la loi de Zipf inverse.....	102
5.3	Contraste et gradient.....	104
5.3.1	La loi de Zipf.....	104
5.3.2	La loi de Zipf inverse	106
6	Applications	109
6.1	Les lettrines.....	110
6.1.1	Moteur de recherche de lettrines	115
6.1.2	Évaluation.....	117
6.1.3	Évaluation par les histogrammes.....	123
6.2	Les manuscrits.....	125
6.2.1	L'indexation d'images de manuscrits.....	126
6.2.2	Identification de scripteur par la loi de Zipf	127
6.2.3	Évaluation.....	131
6.2.4	La loi de Zipf inverse et les manuscrits.....	132
6.3	Les images couleur	135
6.3.1	Évaluation.....	136
6.3.2	Conclusion.....	137
7	Conclusion & perspectives	139

INTRODUCTION

1. Introduction

1.1 Contexte et motivations

Depuis la nuit des temps l'homme n'a eu de cesse de vouloir acquérir des savoirs nouveaux et des expériences, mais pas seulement pour son propre usage. Il a toujours voulu en faire bénéficier les générations futures. Ainsi il a été animé par une volonté de transmettre à ses descendants les choses bonnes aussi bien que les mauvaises, c'est à dire celles considérées comme nécessaires à la survie de l'espèce et celles nocives à sa pérennité bien que bénéfiques pour le développement égoïste et considérées comme contraire à une "morale" communément admise. A une époque où l'écriture n'existait pas encore, cette transmission s'est faite principalement par la transmission orale mais aussi par l'image, en effet les fresques murales des grottes préhistoriques sont une illustration de ce processus. Au cours des siècles les dessins tracés par l'homme se sont transformés en symboles souvent d'ailleurs issus de la stylisation de pictogrammes. Avec l'introduction de l'écriture, les textes ont progressivement remplacé les dessins. Jusqu'à encore récemment, le texte était à la base de l'éducation, de l'enseignement et de l'échange d'information mais avec l'arrivée des nouvelles technologies, la communication et l'éducation utilisent et réutilisent de plus en plus les images. Plusieurs avancées technologiques sont à l'origine de cette évolution. Les logiciels grand public permettent la création, la reproduction aisée d'images et le développement des mémoires rendent possible le stockage d'une grande masse de données et les réseaux disponibles avec un bon débit suscitent de nombreux échanges de texte mais aussi d'images, bien plus volumineuses en bande passante que les textes.

Cette masse d'information n'aurait aucun intérêt si l'on ne pouvait facilement retrouver les

documents concernant un centre d'intérêt particulier. Cela justifie que l'indexation d'image soit un domaine en pleine expansion, c'est un vaste domaine en raison de la variété des documents qui sont disponibles mais aussi en raison de la variété des types de questions que l'on peut se poser. Au delà des nécessités liées à l'enseignement et au partage du savoir, l'explosion des nouvelles technologies, l'évolution des réseaux et l'accès à Internet haut débit pour tous, ont conduit les flots multimédia à ne pas cesser de croître ces dix dernières années. La part grandissante des flux multimédia dans les échanges de données amène de nouvelles problématiques quant au classement et à l'indexation de ces flux. Là où les outils d'indexation de textes sont nombreux, mais toujours très performants, les outils d'indexation multimédia bien qu'en pleine effervescence n'en sont qu'à leur début. L'image est devenue la composante essentielle de ces flux multimédia, on pourrait presque parler d'unité élémentaire multimédia. On doit constater que beaucoup d'outils d'indexation d'image se bornent à l'analyse de métadonnées souvent textuelles introduites de manière manuelle, pour indexer et classer les images. C'est le cas de Google qui utilise le contexte d'une image (son nom, les mots contenus dans la page web hébergeant l'image...) pour la classer.

L'image, pourtant, est bien souvent, plus riche que le texte, en contenu, moins subjective et surtout indépendante de la langue, même si son interprétation n'est pas indépendante de la culture de celui qui reçoit l'information. Ne dit-on pas qu'un dessin vaut mieux qu'un long discours. En revanche, si elle renferme plus d'information que le texte, l'image se révèle occuper plus de place que ce dernier quand on compare les tailles des fichiers contenant une page de texte et une page d'image. Ce volume occupé par l'image est problématique, tant pour le stockage de celle-ci que pour son classement. Un support numérique contenant des images mises pêle-mêle, n'est pas d'une grande utilité car l'information ne pouvant facilement être retrouvée est comme perdue. De tout temps les humains ont cherché à indexer les documents pour mieux les retrouver. Néanmoins, les classements humains sont souvent subjectifs dans le choix des index, et fastidieux à mettre en œuvre. On peut affirmer que tout stockage d'une masse de données nécessite un système d'indexation robuste et rapide permettant, dans le cas des images, une recherche par similitude visuelle.

Pour le monde du document, et surtout du document ancien, la numérisation de masse est aujourd'hui vitale, en effet non seulement elle permet de pouvoir travailler sur des copies digitales et ainsi de préserver des documents déjà bien abîmés par le temps, mais elle permet aussi de pouvoir partager de façon rapide et distante un grand volume de documents mis à la disposition d'un public très varié, chercheurs de différentes disciplines aussi bien que public curieux. Cette numérisation qui permet de stocker un nombre impressionnant de documents sur des supports numériques fait apparaître tout de même un inconvénient majeur. En effet le temps de numérisation est déjà long mais

l'indexation par métadonnées ou l'étiquetage manuelle se révèle être un travail titanesque. Des outils automatiques d'indexation par le contenu sont devenus incontournables dans le domaine de la numérisation des documents anciens. De plus, les mauvais choix stratégiques dans la politique de numérisation peuvent avoir des conséquences dramatiques et réduire à néant des années de numérisation. En effet une numérisation peut être suffisante pour une utilisation par l'œil humain, mais se révéler insuffisante pour une automatisation des tâches par l'ordinateur. Le temps est précieux et souvent la numérisation de document est une course contre la montre pour sauver des documents qui se dégradent à vue d'œil, ou encore ne pas se laisser déborder par les nouvelles acquisitions. Il est souvent hors de question, pour des raisons financières, de recommencer la digitalisation et donc il faut trouver des méthodes d'indexation et plus généralement de traitement d'image adaptées à cette mauvaise stratégie. On ne peut plus parler aujourd'hui de sauvegarde de patrimoine sans parler de numérisation et donc d'indexation.

La protection des documents anciens contre les effets du temps n'est pas le seul problème que permet de résoudre la numérisation. Le document ne se cantonne pas à n'offrir qu'un simple texte à lire et à interpréter, la composition du document, sa structure, sa forme, ses enluminures contiennent aussi de riches informations. Même avant l'invention de l'imprimerie les livres anciens ont toujours contenu de petites illustrations servant à décorer ceux-ci. Au début, peintes à la main, ces décorations ne sont pas simplement là pour rendre plus attractive la lecture du livre, elles sont souvent en adéquation avec le sens du texte. Prenons l'exemple particulier des lettrines, celles-ci, avec le temps, ont fini par être plus qu'une simple lettre posée sur un entrelac de végétations. Petit à petit les dessins servant de fond à la lettre représentèrent de véritables scènes de vie avec de petits personnages, des paysages, des animaux. Ces scènes étaient une illustration de ce que l'on va trouver dans le paragraphe débutant par la lettrine. Plus encore l'analyse informatique de ces lettrines va permettre de trouver des informations cachées. En effet avec l'avènement de l'imprimerie, les lettrines sont obtenues à partir de petits tampons gravés par un artisan. On trempait le tampon dans l'encre pour transférer l'image sur le document à imprimer. Un jeu de tampons était cher à réaliser et économiquement il est peu probable qu'un jeu ne serve que pour une série d'un même livre. Les tampons ont pu être vendus ou échangés, les enlumineurs ont pu être embauchés chez d'autres éditeurs, certains ont pu copier les lettrines d'autres, les tampons, à force d'être utilisés, se sont dégradés, se sont cassés. La trace de cette dégradation apparaît sur les livres et laisse ainsi une information de l'histoire de ces péripéties sur les livres qu'ils illustraient. L'observation de ces variations dans les détails n'est possible dans la masse des documents que grâce au traitement de l'image. Une indexation, en particulier, va permettre de trouver des similarités entre des lettrines

venant de documents différents et donc de retracer leurs histoires. Cette recherche sur documents numérisés et dont le partage est de l'ordre du possible accélère les découvertes des chercheurs en sciences humaines, la recherche sur différents sites où se trouvent les originaux est un travail d'historien qui demande de longues heures de voyage et de lecture qui se trouvent remplacées par un travail sur Internet. Par exemple, le regroupement de lettrines assez semblables pourrait même conduire à retrouver des informations perdues sur un document. Souvent les livres sont très dégradés et on ne peut plus savoir avec certitude d'où ils viennent, qui en est l'auteur ou l'imprimeur. Le traitement informatique peut, en analysant les images, constituer une aide pour retrouver ces informations perdues. Toutes ces informations cachées à différents niveaux d'un document induisent que lorsque l'on cherche à indexer des images, on le fait de façon adaptée à ce que l'on veut retrouver. En reprenant l'exemple de nos lettrines, une indexation de celles-ci pour les classer en fonction de leur similarité globale, ou en fonction de la lettre qu'elles symbolisent ou encore pour trouver l'usure du tampon, mène à des méthodes d'indexation bien différentes. Deux niveaux se dégagent, le premier que l'on qualifiera de global pourrait être comparé à ce que fait l'humain pour dire si, au premier coup d'œil, deux images se ressemblent ou non. Cette méthode se doit d'être simple et doit prendre en compte l'ensemble de l'image. Beaucoup d'études ont montré que l'homme était capable, jusqu'à un certain point, de discriminer et d'indexer les objets de manière globale, en faisant travailler la partie gauche de son cerveau [VIN03]. On sait que l'homme peut dénombrer les groupes d'objets jusqu'à cinq sans être obligé de compter ceux-ci, d'un seul coup d'œil nous pouvons donner le nombre d'objets mis sous nos yeux si celui-ci est inférieur ou égal à cinq. En revanche dès que ce nombre passe la barre de cinq, nous devons compter les objets un par un pour pouvoir en donner leur nombre. En prenant appui sur ces constatations nous appellerons méthodes globales les méthodes syncrétiques. Les méthodes s'appuyant sur une analyse plus fine et plus détaillée des images ou de parties de ces images s'appelleront des méthodes locales. On voit se dessiner ici deux approches différentes et qui ne peuvent pas avoir pour objectif de répondre aux mêmes interrogations.

L'autre enjeu dans l'indexation d'images est de savoir si l'on veut calquer ou non le comportement humain. En effet pour évaluer un système d'indexation on se réfère à notre propre sens, du moins au sens d'un expert du domaine. Force est de constater que ceci est non seulement souvent inadapté mais aussi subjectif car la variabilité des avis des experts est souvent non négligeable. Des images venant de bases de données totalement indépendantes ne seront pas obligatoirement indexées de la même manière par deux personnes distinctes. Plus encore, une même personne ne réalisera pas non plus la même indexation à deux moments différents. Les contextes sociaux, culturels et temporels auront un impact énorme sur les résultats. Cela n'est pas inhérent uniquement

à l'indexation d'images, on retrouve ce phénomène dans l'indexation de texte ou de tout autre matériau. Prenons l'exemple d'un moteur de recherche d'images, transposé de Google textuel, celui-ci nécessiterait une requête pour trouver les similarités. Imaginons que cette requête ne soit pas textuelle mais sous forme d'une image dite image requête [VAL06]. Par exemple quel pourrait être le résultat d'une image requête représentant un coucher de soleil sur une plage peuplée de palmiers ? Cherche-t-on toutes les images qui présentent un coucher de soleil, celles qui représentent de la végétation exotique, celles qui ont une nature marine ou les trois à la fois. De plus, peut-on et comment, affiner ses recherches en fonction des premiers résultats ? A priori dans un moteur de recherche textuel [LEL99], un protocole communément utilisé pour trouver les informations pertinentes dont on a besoin, est de commencer par composer une requête très lacunaire, pour, petit à petit, en fonction des résultats, la préciser avec des termes orientant le moteur de recherche positivement et négativement en fonction de nos désirs et des résultats déjà proposés. Comment retranscrire cette méthode dichotomique dans le domaine de l'image ? Les moteurs de recherche d'images existant ne proposent pas actuellement tous ces mécanismes. Même si depuis quelques temps certains moteurs comme celui de l'Inria nommé lmedia [FAU04] essaient de mettre en place un feed-back ils perdent vite de leur intérêt quand on se rend compte qu'ils ne s'appuient plus lors du feed-back que sur les images résultat et non sur ce que cherche l'utilisateur. Ce que l'on aimerait trouver c'est une analyse en fonction des différentes requêtes que passe un utilisateur, de ce qui l'intéresse dans la recherche, de quels sont ses critères d'efficacité qu'il privilégie. Que dire aussi sur les moteurs de recherche d'images basés sur des recherches par le contenu mais dont les requêtes sont textuelles [FRA96].

A partir de ces constats, de nombreuses méthodes d'indexation que nous explorerons ultérieurement ont été développées. Celles-ci sont souvent adaptées à des catégories d'images bien précises et aux informations que l'on recherche. L'inconvénient majeur est que pour développer un moteur généraliste d'indexation d'images, il faut analyser les images et utiliser des méthodes ou des critères différents en fonction des types d'images à traiter. C'est ainsi qu'il devient crucial de trouver une méthode ou un ensemble de propositions, applicable à toutes les images, et qui ne nécessiterait qu'un simple paramétrage dans les primitives issues de l'indexation pour répondre à une requête. C'est pourquoi il nous a semblé préférable de privilégier dans un premier temps les méthodes dites globales qui donneront des résultats de similitude proches de ceux qu'un humain aurait trouvés. Celles-ci sont bien adaptées à un mode d'interrogation d'un moteur d'indexation qu'est le sketch [BRO05]. Ce mode demande à l'utilisateur de présenter comme requête une ébauche grossière visuelle de ce qu'il cherche. Par exemple s'il cherche des images de coucher de soleil sur la mer il pourra, avec un outil rudimentaire de dessin, tracer un rond jaune orangé surplombant une zone plus ou

moins rectangulaire bleue. En effet, on peut à la limite dire qu'une image est une somme de contours, de couleurs et de textures.

1.2 Contributions et organisation du rapport

Ce document va s'articuler en quatre grandes sections. Dans la première partie nous allons établir un état de l'art de ce qu'est aujourd'hui l'indexation d'image et quelles sont les méthodes les plus utilisées et reconnues dans ce domaine. Nous passerons en revue les différentes méthodes et nous regarderons quels sont leurs domaines d'efficacité. Cette même partie sera composée de deux sous sections classant les méthodes en deux groupes celui des méthodes dites « locales » ou « structurelles ». Cette partie permet de replacer notre travail au sein des développements récents.

Ensuite nous précisons l'outil de base sur lequel s'appuie notre méthode, la modélisation par des lois puissance et plus particulièrement les lois de Zipf. Ces bases nous permettront dans les deux chapitres suivants de développer notre approche.

La dernière partie présentera plus en détails la mise en œuvre et les résultats de deux applications que nous avons développées.

ÉTAT DE L'ART DE L'INDEXATION D'IMAGES

2 État de l'art de l'indexation d'images

Dans cette section nous allons définir ce que nous entendons par l'indexation d'images puis nous analyserons les méthodes les plus marquantes permettant de réaliser cette indexation. Deux groupes de méthodes d'indexation vont se dégager. Le premier contient les méthodes qui consistent à cumuler des informations extraites de l'image pour caractériser une image puis à comparer les informations associées à deux images pour connaître le niveau de similarité entre les deux images. Le deuxième, quant à lui n'utilise que le contenu de l'image pour l'indexer. Nous verrons deux classes de méthodes d'indexation par le contenu, l'une utilisant l'image dans sa globalité et l'autre misant sur des approches plus locales se basant sur l'extraction de parties de l'image.

2.1 Définitions

Dans un premier temps, et pour une meilleure compréhension de ce qui suit, il semble nécessaire de définir ce qu'est une image. Cela peut paraître trivial mais la structure particulière de cette composante multimédia va avoir un impact prépondérant sur la manière de l'indexer.

En effet une image est un ensemble de valeurs dans un espace colorimétrique donné. En général on utilise l'un des deux espaces colorimétriques les plus usuels, un espace monodimensionnel (image en niveaux de gris) et un espace de dimension trois (espace couleur). Dans la pratique une image est représentée par une matrice à deux dimensions dont les éléments sont constitués par une quantification de l'état des pixels, unité élémentaire d'une image numérique. Un pixel est une surface élémentaire caractérisée le plus souvent par une valeur qui mesure son niveau de gris qui peut varier de 0 à 255,

cette valeur est stockée sur un octet. En couleur, au format dit rouge vert bleu (RVB), un pixel est caractérisé par un ensemble de trois valeurs comprises entre 0 et 255, soit trois octets. Une image couleur de 800 sur 600 pixels a un volume de 1,44 Megaoctets. Son organisation spatiale en matrice à deux dimensions et le volume important qu'elle occupe en taille vont rendre son indexation plus épineuse que celle d'un texte numérique qui demande moins de temps de calcul pour être classé.

Dans la suite nous considérerons une image comme un ensemble de pixels en 256 niveaux de gris ou dans un espace colorimétrique Rouge Vert Bleu, et qui sont liés par une relation de voisinage.

A présent nous allons essayer de dégager une définition de l'indexation et plus spécifiquement de l'indexation d'images. Si l'indexation de texte semble un concept familier celui relatif à une image est un peu plus inhabituel.

L'indexation est le fait d'ajouter des valeurs à un ensemble d'informations qui est dit indexé. L'indexation consiste à résumer une chose par des éléments caractéristiques issus de cette chose en vue de pouvoir la classer et la rapprocher des choses similaires.

La première mise en application de l'indexation fut certainement lors de la constitution des bibliothèques. En effet avec la croissance du nombre des ouvrages à conserver, la nécessité de résumer leur contenu en vue de pouvoir interroger le bon ouvrage en fonction des volontés des usagers devint essentielle. L'indexation d'un corpus textuel consiste à repérer dans celui-ci les mots et expressions significatifs dans un contexte particulier et à créer un lien entre ces termes et le texte original.

Le système textuel d'indexation le plus répandu à l'heure actuelle est sans doute celui du répertoire téléphonique où les contacts sont classés en fonction de la première lettre de leur nom. L'indexation permet de grouper des valeurs en vue de les retrouver plus rapidement. Les premiers systèmes d'indexation numérique furent textuels, partant du simple agenda électronique pour arriver aujourd'hui à l'apothéose du domaine qu'est Google. De nombreuses études ont été réalisées sur l'indexation de texte [FAR96], [WEC97]. Avec l'arrivée des nouvelles technologies, l'indexation est devenue indispensable dans le monde de l'informatique. Le logiciel élémentaire à tout ordinateur qu'est le système d'exploitation ne repose que sur des mécanismes d'indexation de fichiers et depuis les tous premiers systèmes UNIX ou DOS, l'indexation de fichiers fut présente au cœur du noyau de ceux-ci. Au début, ce mécanisme était considéré comme un simple moyen d'accélérer les recherches dans la table d'allocation des fichiers, l'indexation est aujourd'hui la pierre angulaire de tout système d'information. Inutile de préciser qu'une base de données sans système d'indexation devient totalement inutilisable avec l'accroissement du nombre d'enregistrements qu'elle contient. Dans le

monde informatique, c'est dans les systèmes de gestion de bases de données relationnelles que l'indexation a connu son premier essor, bien sûr dans le but d'accélérer les recherches, mais aussi de permettre dans les systèmes les plus gros, l'équilibrage des informations stockées sur les différents supports. Aujourd'hui certaines bases de données se voient obligées, vu leur volume, d'être stockées sur plusieurs disques. Les données sont donc morcelées et réparties, et dans un but d'utilisation uniforme des supports elles doivent occuper à peu près le même volume sur chacun des disques [TIM88]. L'indexation est tellement importante qu'il n'est pas rare de voir des bases de données stockant les mêmes informations coexister uniquement parce qu'elles ne sont pas indexées de la même façon. L'exemple le plus connu est celui des annuaires type pages jaunes qui sont indexés pour optimiser une recherche nom vers téléphone et qui rendent une recherche inverse fastidieuse et ultra coûteuse en temps. Il a fallu créer une base redondante mais indexée pour accélérer les recherches de type téléphone vers nom.

Ces exemples montrent que l'indexation en informatique est aujourd'hui la clef de voûte de tout système dit d'information qu'il soit managérial ou décisionnel. La recherche d'informations est devenue vitale et l'essor de Google n'en est qu'une conséquence directe. L'indexation est peut être le mécanisme le plus vieux qui soit, les bibliothèques de tout temps y ont eu recours, et il semblerait que même notre cerveau soit pourvu d'un système permettant de ranger les souvenirs en fonction de ce qu'ils nous concernent directement ou non, qu'ils soient à long terme ou non. De nombreuses études montrent la relation entre l'anatomie du cerveau et la mémoire [MEU03]. Il est même concevable que quand on a du mal à retrouver une information, le phénomène que tout le monde a connu, quel est le nom de cet acteur que je vois, le nom de cette chanson que j'entends, cela est dû à une organisation des données dans le cerveau inadaptée à notre recherche présente, fonction des données perçues. Il semblerait que même les neuromédiateurs, les synapses et les neurones ne soient pas organisés pêle-mêle dans notre tête, ils sont sûrement indexés de manière à nous donner les informations, le plus rapidement possible. Les liaisons ne semblent pas être construites selon un unique schéma d'apprentissage puisque les recherches s'effectuent selon les individus plus ou moins de manière globale, comme le montrent les études sur la reconnaissance de visages [FIO94a].

2.2 Indexation par les métadonnées

Avant de donner un aperçu des différentes méthodes d'indexation par métadonnées, rappelons ce qu'est une métadonnée.

Tout ensemble structuré d'informations concernant une image comme son nom, sa

provenance, sa date, sa qualité et la description de ce qu'elle contient pour sa mise en relation avec les autres images de la collection sont des métadonnées. On peut dire qu'une métadonnée est une donnée à propos d'une donnée. Les métadonnées ont deux facettes, elles doivent s'attacher à décrire le contenu d'une zone de l'image et elles formulent comment cette information doit être annotée d'un point de vue technique. Les métadonnées sont de plusieurs types : Les métadonnées descriptives dont le rôle est d'informer sur ce que l'on voit dans telle ou telle partie de l'image.

- Les métadonnées administratives qui donnent des informations qui ne sont pas dans l'image (titre, date ...)
- Les métadonnées de propriété intellectuelle qui gèrent les droits de l'image à être utilisée par un tel ou un tel.
- Les techniques qui informent sur les propriétés physiques de l'image

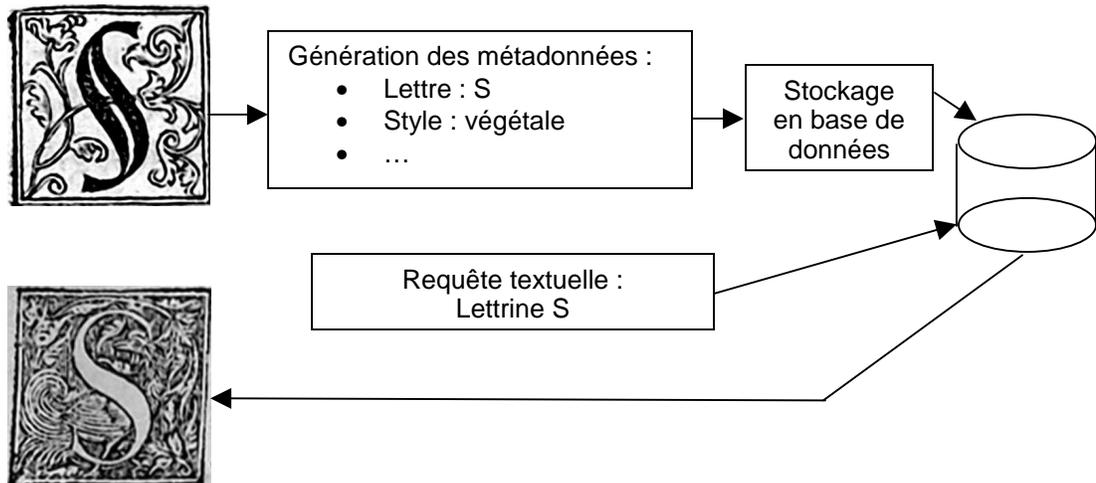


Fig 2.2.a : Exemple de système de recherche d'image par les métadonnées

L'indexation par métadonnées peut prendre plusieurs formes, la première, la plus simple techniquement à mettre en œuvre est de faire étiqueter manuellement les images par des humains. Ainsi c'est un groupe de personnes qui va devoir décrire ce que représente l'image. Cette façon d'indexer a de gros avantages au niveau technique, elle ne demande que peu de moyens informatiques, une simple base de données suffira pour héberger la totalité du moteur de recherche. De plus, dans le cas d'une indexation orientée, c'est-à-dire où l'on sait quel type d'information est recherché dans les images, ces méthodes vont être très efficaces. En revanche beaucoup d'inconvénients apparaissent avec ces méthodes, le premier étant la subjectivité humaine, en effet dans un système d'indexation généraliste deux humains ne donneront pas les mêmes métadonnées pour qualifier une même image, les zones d'intérêt d'une image varieront

en fonction du passif de chacun, de son contexte culturel. De plus, pour être efficace un moteur d'indexation ne peut contenir un volume énorme de métadonnées, il faut limiter la description de l'image. On peut signaler dans ce domaine la constitution d'Iclass par les historiens pour l'indexation d'images de tous ordres [COU78].

Des systèmes de création automatique de métadonnées ont été décrits [MOR04]. Dans ces systèmes on va essayer de créer automatiquement des informations textuelles extraites soit du contexte de l'image, par exemple les mots clés du site Internet où l'image se trouve, soit des propriétés techniques de celle-ci.

Qu'elle soit automatique ou non, l'extraction de métadonnées se doit d'être accompagnée par un format dans lequel on va stocker les métadonnées. Au cours des temps, les systèmes proposés ont évolué. Au début, les structures étaient propriétaires et donc incompatibles entre elles, mais petit à petit des systèmes, se disant universels, de stockage de métadonnées sont apparus [WEI98-DEM96]. Aujourd'hui le langage XML a été pris comme support et, grâce à sa grande adaptabilité, il est parfait dans l'exercice de stockage des informations sur une donnée multimédia.

Même si les métadonnées semblent être matures aujourd'hui dans le domaine de l'indexation d'images et même si certains scientifiques voient en elles une chance pour une meilleure interopérabilité, il reste encore beaucoup d'inconvénients à utiliser ces méthodes. Premièrement, les métadonnées sont chères non seulement en argent mais aussi en temps. En effet la création de métadonnées n'est pas une opération triviale, elle demande de mobiliser du temps et des personnes. De plus, un second inconvénient est la complexité des outils existant de stockage de ces métadonnées, ce stockage représente souvent un investissement assez important. Le troisième inconvénient provient de la subjectivité des métadonnées comme nous le disions un peu plus haut. Un autre défaut de ces méthodes est le côté infini des métadonnées, en effet la description à l'aide de mots clés d'une image peut varier de façon quasi infinie, la liste n'est jamais exhaustive. Les métadonnées ont un nombre grandissant d'adversaires qui trouvent leur utilisation complètement obsolète [DOC01]. En effet avec l'évolution des méthodes d'indexation par le contenu les métadonnées deviennent de moins en moins utiles. On leur préfère de plus en plus des techniques qui reposent, au moins dans une première étape, sur l'image seule, c'est l'indexation par le contenu.

2.3 Indexation par le contenu

La première question que l'on pourrait se poser quand on pense à l'indexation et à la recherche d'images par le contenu c'est de savoir si le but reste le même que pour l'indexation textuelle. Une différence évidente est que la manipulation d'un flux en deux dimensions nécessite des méthodes plus complexes qu'avec un signal

monodimensionnel. De plus, l'homme est depuis longtemps familiarisé avec la recherche textuelle, on classe des textes par ressemblance depuis la nuit des temps avant même l'apparition des premières bibliothèques, c'est la détection de la thématique. La recherche d'images à partir d'une image, entendez que pour avoir un résultat il faut fournir au système une image requête, n'en est qu'à ses débuts. Si l'on comprend vite ce qui va nous permettre de rechercher des images et ce qui va nous donner une idée du taux de similarité dans une indexation utilisant des métadonnées, ceci est un peu plus abstrait dans l'indexation par le contenu car la qualité des résultats est difficilement quantifiable tant il y a de subjectivité dans le jugement humain. Dans notre cas, nous voulons extraire des images, des valeurs qui vont caractériser ces images. Ces valeurs sont calculées mathématiquement à partir de l'analyse des pixels de l'image. On nommera ces valeurs des primitives ou descripteurs ou index. Nous reviendrons sur ces notions un peu plus loin.

Un autre trait caractéristique de l'indexation par le contenu que nous voulons introduire est la frontière entre images identiques et images similaires. Les moteurs de recherche informatique textuelle les plus rudimentaires se bornent généralement à essayer de retrouver une phrase dans un ensemble d'ouvrages. Par exemple, l'utilisateur veut trouver le livre dans lequel le héros dit : « Pour être dévot, je n'en suis pas moins homme ». Il y a une idée bien précise et unique de ce que l'on cherche, et pour la décrire, rien de plus facile, il suffit de l'écrire et de la rechercher telle qu'elle, aux erreurs typographiques près. Cette facilité n'est pas présente dans le monde de l'image. En effet, nous n'avons pas toujours la même façon de visualiser ce que l'on cherche, ai-je quelque chose de très précis en tête quand je cherche de belles voitures rouges ? De plus les déformations sont légions dans les images. Imaginons que l'utilisateur cherche un visage bien précis dans un ensemble de photos de groupe, mon image requête sera par exemple une photo d'identité. Il serait peu probable que le visage de la personne se trouve représenté exactement de la même façon dans les photos de groupe que sur la photo d'identité. Pour simplifier, un texte n'est pas soumis aux mêmes déformations qu'une image, la requête textuelle est exactement identique à ce que l'on trouvera dans l'ouvrage. Même si cette comparaison est un peu simpliste, elle permet de mieux comprendre les contraintes qui pèsent sur l'indexation d'images.

De plus, notre vocabulaire dans le domaine de l'image est potentiellement infini. On entend ici par vocabulaire l'ensemble des éléments composant un langage. On peut affirmer que l'image est le support d'un langage dont le vocabulaire est infini. En effet, je n'ai que peu de mots pour décrire un objet dans le langage parlé, il y a peu de synonymes au mot pomme, en revanche j'ai une infinité de façons de représenter une pomme par une image. On pourrait avancer que le vocabulaire du langage est fermé, donc énonçable en extension alors que la nature nous donne une variété infinie

d'occurrences des objets qu'elle génère. C'est pourquoi si dans l'indexation de texte ou même dans l'indexation d'images par métadonnées, on cherche à trouver des choses plutôt identiques, en indexation d'images par le contenu nous parlerons de choses similaires. C'est d'ailleurs à cause de cette notion assez subjective de similarité que l'indexation d'images par le contenu n'est pas une simple adaptation des méthodes textuelles au domaine de l'image.

Si la notion de vocabulaire infini est un problème, le format de nos données en est un autre. Comme nous l'avons rappelé plus haut l'image est une matrice de pixels en deux dimensions, mais ce qu'elle représente peut varier énormément d'une image à une autre. En effet nous pouvons être confrontés à des prises de vue numériques d'œuvres différentes :

- de dessins
- d'estampes (gravures, lithographies...)
- d'enluminures de manuscrits (bandeaux, lettrines ...)
- de textes manuscrits ou non, numérisés
- de sculptures, peintures, objets muséaux
- de photographies
- de cartes postales, plans d'architectes...
- ou d'images produites directement en numérique

Il est à noter que, à taille fixée, le nombre des images digitalisées possibles est fini puisque le nombre de pixels est fixé ainsi que le nombre de valeurs caractéristiques associées à chaque pixel. La grande variété de types d'image, corrélée au fait qu'une image peut être en noir et blanc, en niveaux de gris ou en couleur va avoir un impact indéniable sur son indexation.

Après avoir vu les caractéristiques de l'indexation d'image par le contenu intéressons nous à sa mise en pratique. On pourrait définir l'indexation d'image par la nécessité de synthétiser une image en vue d'une future mise en correspondance. En effet, le but de l'indexation est la recherche d'images similaires. On comprend assez vite comment marche un moteur de recherche d'images basé sur les métadonnées, c'est le cas de Google où c'est une requête textuelle qui va rendre comme résultat une série d'images classées par pertinence. Ici la requête n'est plus un texte, mais une image [BOU01] que l'on va soumettre au système. La figure ci-dessous montre comment on peut représenter un moteur de recherche d'images.

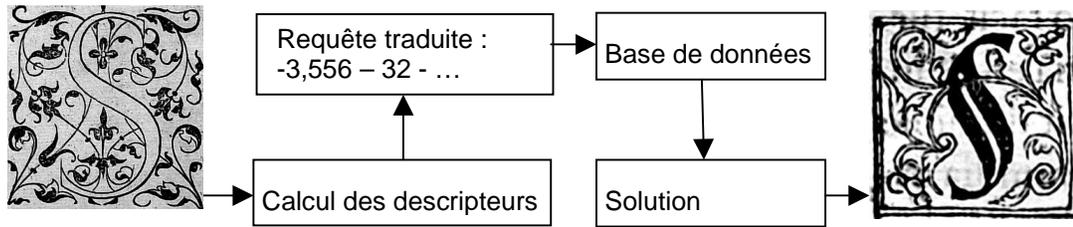


Fig 2.3.a : Comportement d'un système de recherche d'images à partir d'une image

Pour accélérer la recherche on pourra par avance indexer les images de la base de données et ne conserver que leurs descripteurs.

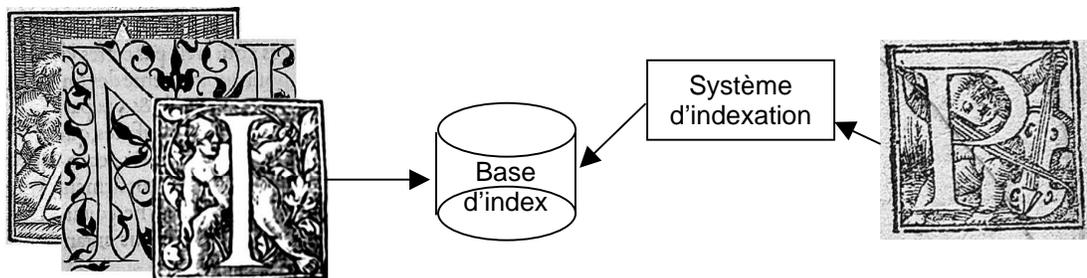


Fig 2.3.b : Comportement d'un système de recherche d'images pré indexées

Comme nous l'avons dit plus haut, pour indexer des images il faut en extraire l'information pertinente. Si cette phase semble facile à réaliser en matière textuelle, elle est un peu plus abstraite en ce qui concerne l'image. En effet les descripteurs d'une image doivent être concis, discriminants et adaptés à la mise en correspondance. On pourra caractériser une image selon plusieurs points de vue, la couleur, la texture ou la structure. Deux façons d'indexer les images vont donc se détacher, la première est une approche structurale de l'image qui s'intéresse aux zones et aux objets composant l'image, l'autre est une approche statistique qui traite plutôt l'image comme une globalité.

2.4 Approche structurale

Les méthodes qui relèvent de cette famille partent d'un constat : les images sont formées d'un ensemble de régions spatialement réparties. Cette répartition n'est pas faite de façon aléatoire, elle est organisée de façon à représenter le plus fidèlement possible ce que voit l'œil humain [STE92]. Ces méthodes vont s'attacher à comparer, par exemple, des histogrammes locaux de couleur pour savoir si deux images se ressemblent ou non [ENN95]. D'autres approches vont utiliser des graphes et des arbres couvrant comme descripteur d'une image [SUR05].

Les approches structurales s'appuient souvent sur une segmentation pour pouvoir

indexer une image [BLO99]. Ainsi l'extraction des régions est une des premières orientations adoptées par les approches structurales, en effet une possibilité pour étudier la similarité entre deux images est de rechercher les objets qui la composent pour pouvoir les retrouver dans d'autres images [SAL00]. Ces méthodes, si elles semblent très prometteuses, intelligentes et peut-être d'un prime abord proche de ce que fait l'humain, restent encore imparfaites. Des études en perception visuelle ont montré que notre analyse de la structure d'une image dépend de la distance, du temps et du contexte au moment de la vision de l'image. Ainsi il est démontré que notre analyse spatiale humaine dans des conditions d'observation non optimum n'était sensible qu'aux fréquences spatiales faibles. Ainsi si nous nous plaçons devant l'image, à une distance réduite, en prenant notre temps pour la regarder nous aurons tendance à nous soucier des hautes fréquences spatiales. C'est ce qui se passe avec le portrait d'Abraham Lincoln de Salvador Dali.



Fig 2.4.a : Lincoln peint par S.Dali

Cette image est intéressante car elle permet de voir la différence entre un traitement global ou statistique et un traitement plus structurel et local. Si nous prenons en compte uniquement des primitives globales, cette image est proche d'un portrait de Lincoln en revanche, si l'on s'attache un peu plus aux zones qui composent l'image nous voyons apparaître une femme peu vêtue.

Beaucoup de méthodes de segmentation d'image ont été trouvées et décrites, mais aucune ne semble être totalement généraliste. On peut même se permettre un simple constat, il est clair que la segmentation d'image n'est pas uniquement un problème de traitement informatique de l'image mais surtout aussi il est lié au contexte culturel de la personne qui segmente. En effet pour pouvoir bien segmenter les objets dans une image, il faut avoir des connaissances de base a priori sur cet objet. On dit que les

méthodes de segmentation sont efficaces dans un environnement non naïf.

Les approches structurales ont de nombreux défauts, en premier, la notion de régions dites homogènes est très subjective, il est facile de découper des objets dans une image de type dessin, mais dans une image fortement structurée, l'opération est plus délicate. Que signifie homogénéité lorsqu'une image représente un arlequin ? De plus, les graphes obtenus sont souvent très instables d'une image à l'autre, et de surcroît très sensibles à la moindre déformation de l'image.

2.4.1 Segmentation par fenêtre

Une technique simple pour segmenter une image et avoir une approche plus locale de l'indexation est d'utiliser un découpage en grille de celle-ci. Chaque case sera donc traitée indépendamment lors de l'indexation.



Fig 2.4.b : Segmentation d'image par une grille

Un avantage de cette méthode est sa simplicité et sa rapidité. En revanche le choix de la taille des cases de la grille est délicat et est un point crucial de la méthode. La méthode est sensible à un changement d'échelle du contenu de l'image. Le terme segmentation est un peu abusif dans ce cas puisqu'il est aveugle, c'est pourquoi les auteurs préfèrent parler de groupement de pixels [FOR03] ou encore de segmentation faible [MED05]. Même si l'idée peut sembler bien loin d'un découpage strict de l'image en objets qu'on nommera découpage sémantique, on préférera un découpage en zones homogènes même comme en figure 2.4.b grossièrement délimitées. Les différents travaux en segmentation de ces dernières années montrent bien que le découpage de l'image en zones sémantiques et la segmentation de bas niveau sont deux choses bien différentes. En effet un objet tel que nous le percevons dans l'image n'est pas forcément homogène

en termes de caractéristiques, tandis qu'une zone homogène de l'image peut très bien contenir plusieurs objets différents.

2.4.2 Segmentation par régions homogènes

Plutôt que d'appliquer une grille aveugle sur l'image pour la segmenter, tout en restant dans la même famille de méthodes, il peut paraître judicieux d'utiliser un découpage plus adaptatif en fonction de l'image et de sa structure.

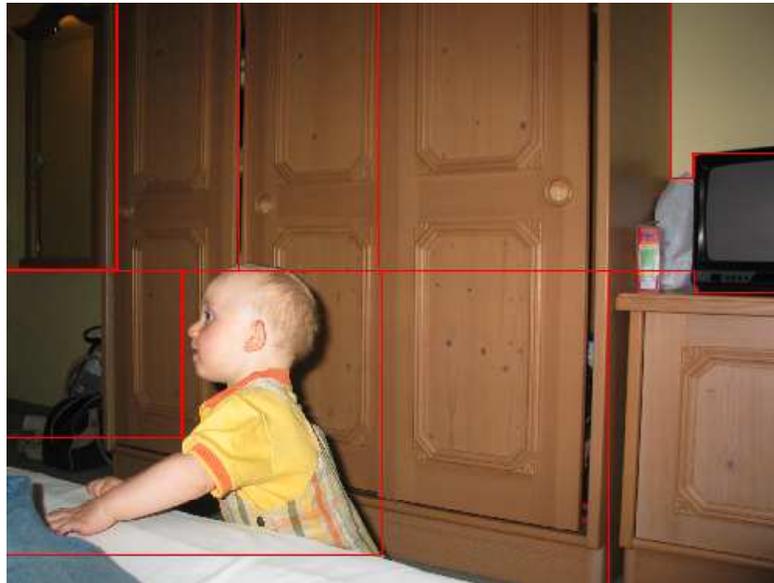


Fig 2.4.c : Segmentation adaptative par régions rectangulaires

La grille va s'adapter au mieux aux différentes régions homogènes contenues dans l'image. Cette méthode semble plus judicieuse, mais elle pose quand même un problème majeur qui est celui du choix des critères d'homogénéité. L'homogénéité peut être au niveau de la couleur, de la luminance, du contraste, du niveau de gris. La taille, la précision de la découpe finale reste aussi un problème lié au choix du seuil où doit s'arrêter le processus récursif de segmentation. Les formes segmentées prennent une forme liée au mode de découpage considéré.

2.4.3 Extraction d'objet par une fenêtre de corrélation

La recherche d'objet dans une image par corrélation consiste à rechercher le centre de gravité d'un objet en testant, en chaque pixel ou région d'une image, la similarité entre ce pixel et le centre d'un modèle de l'objet par l'utilisation d'une mesure de corrélation. Deux pixels sont dits corrélés si leurs valeurs de niveau de gris ou de couleur sont proches au voisinage de ces deux pixels. En effet chercher corrélation entre deux pixels ne serait

pas efficace, c'est pourquoi on va utiliser non pas un pixel mais un ensemble regroupé dans une matrice que l'on nommera fenêtre. On peut voir un exemple de matrices corrélées ou non dans la figure ci-dessous.

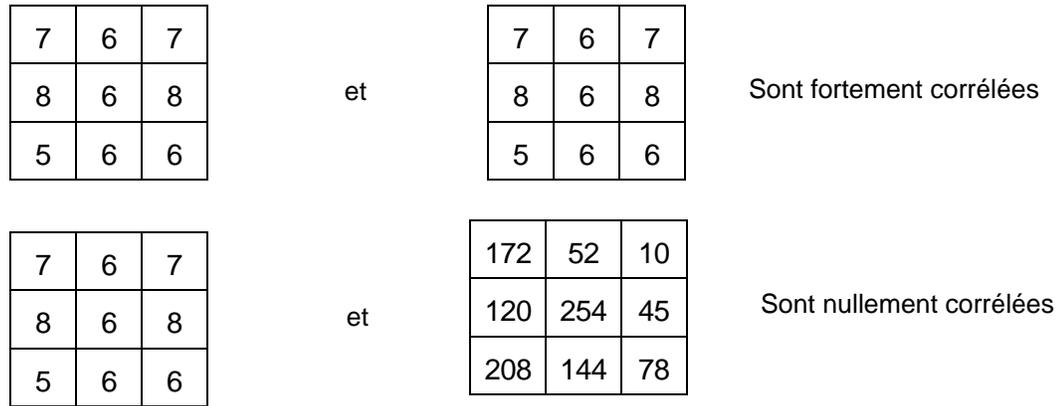


Fig 2.4.d : Exemple de corrélation de matrices

L'observation faite ci-dessus reste vague et très subjective, il est donc nécessaire de trouver des méthodes pertinentes de mesure de corrélation entre matrice.

Il existe des méthodes permettant de ne pas travailler sur des fenêtres rectangulaires, mais celles-ci n'apportent pas une efficacité réellement plus élevée.

La principale difficulté des méthodes basées sur les fenêtres de corrélation réside dans le choix de la taille de la fenêtre de corrélation et dans la mesure de distance de corrélation entre deux fenêtres. Ce type de méthode de segmentation d'image, basée sur les régions et non sur les contours est simple mais la méthode n'est pas invariante au changement d'échelle, à la déformation, aux changements de luminosité et à la rotation. En effet, la taille de la fenêtre d'une image à l'autre devra être adaptée de façon automatique. Le but final est de comparer les fenêtres deux à deux, dans deux images différentes et d'évaluer leur distance/corrélation.

Plusieurs formules de calcul de distances classiques peuvent être utilisées comme la somme des distances ou la somme des carrés des distances :

Respectivement :

$$DI(p_1(u_1, v_1), p_2(u_2, v_2)) = \sum_{i=-N}^N \sum_{j=-P}^P |I_1(u_1 + i, v_1 + j) - I_2(u_2 + i, v_2 + j)| \quad (241)$$

Et

$$D2(p_1(u_1, v_1), p_2(u_2, v_2)) = \sum_{i=-N}^N \sum_{j=-P}^P (I_1(u_1 + i, v_1 + j) - I_2(u_2 + i, v_2 + j))^2 \quad (242)$$

Le principe peut s'appliquer à toutes les distances.

Les limitations de ces approches sont dues aux hypothèses fortes de départ que nous énonçons maintenant

- Les changements de point de vue ne changent pas l'aspect des surfaces. On devrait avoir des surfaces émettant ou réfléchissant un rayonnement dans toutes les directions d'un hémisphère avec une radiance constante. En clair ne sont concernées que des surfaces lambertiennes.
- La forme des régions d'une image à l'autre ne varie pas.
- Deux régions de couleur constante présentent une distance normalisée nulle.

2.4.4 Rétroprojection de primitives

Les parties similaires dans les images candidates sont identifiées dynamiquement : approche souple mais consommatrice en temps de calcul au moment de la requête (par exemple dans VisualSeek [SMI96]). Cette approche qui permet de retrouver et d'extraire dynamiquement des objets d'une image a connu son heure de gloire dans le milieu des années 90. Elle permet en travaillant sur des groupes de couleur d'extraire des zones d'intérêt dans des images en se fondant sur l'hypothèse suivante : une zone d'intérêt ne contient qu'un nombre de couleurs dominantes très faible, et il est facile de les extraire si l'on trouve ce groupe de couleurs pour chaque zone d'intérêt. Il est ensuite possible d'appliquer sur le résultat de la rétroprojection divers filtres en fonction de la nature et de la taille des zones que l'on veut extraire.

La figure 2.4.e donne un exemple de l'application d'une rétroprojection de primitive sur une image cible. On cherche sur cette image à extraire la zone d'intérêt correspondant à l'herbe sèche qui jonche le sol.

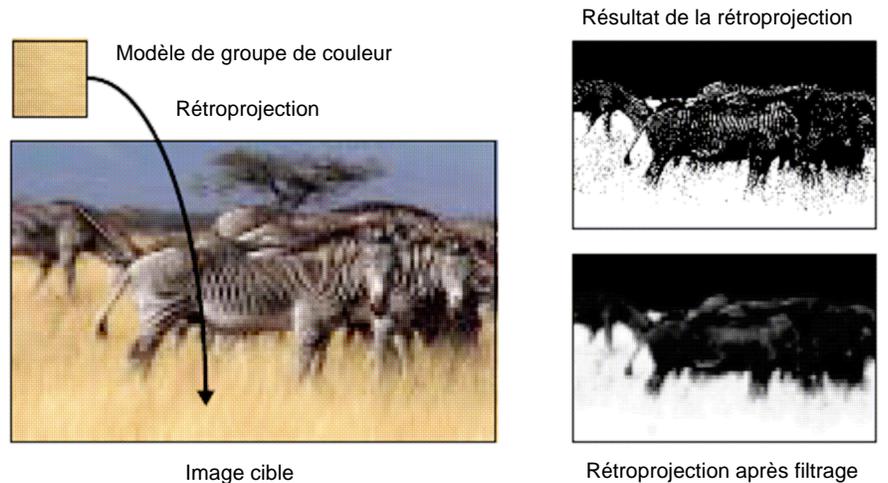


Fig 2.4.e : Exemple d'extraction de zone d'intérêt par rétroprojection issu de VisualSeek

On remarque que l'extraction s'est faite et que la zone en blanc sur l'image résultat après filtrage correspond bien au sol. Cette zone une fois extraite pourra être comparée avec une zone d'une autre image.

2.5 Approche Structurale avec extraction des points d'intérêt

La détection de points d'intérêt est une terminologie récente dans le domaine de la vision par ordinateur. L'extraction de ces points d'intérêts permettra un traitement subséquent des images. Un point d'intérêt est un point particulier d'une image qui peut-être caractérisé comme ceci :

- Il a une définition mathématique légitime.
- Il a une position spécifique au sein de la structure de l'image.
- La structure locale de l'image autour du point est riche en information.
- Le point d'intérêt doit être invariant aux déformations de l'image non seulement en termes de structure et de composantes de bas niveau mais aussi il faut que ces déformations puissent être locales ou globales.
- Et accessoirement, la notion de point d'intérêt doit aussi avoir une composante d'échelle pour pouvoir traiter de façon identique des images naturelles de la même façon avec des résolutions différentes.

La notion de points d'intérêt remonte à la notion de détection d'angle dans la détection d'objet dans les images. Au début, les points d'intérêt servaient dans le domaine de la création assistée par ordinateur pour pouvoir séparer les différents objets 3D contenus dans une image. Rapidement, on s'est rendu compte de son efficacité dans l'indexation

d'image par le contenu. Les méthodes de détection de secteur angulaire se sont vues rejointes par des méthodes de détection de tache (blob). Même si pour certains la notion de détection de blob n'est pas à mettre dans la catégorie de la détection de points d'intérêt, nous ne voyons pas de raison objective de ne pas inclure ces méthodes les unes dans les autres. En termes d'application, on remarque souvent une imbrication entre détection de coin et détection de blob. Aujourd'hui ce sont des méthodes classiques pour extraire des primitives nécessaires à l'indexation et à la recherche d'image.

2.5.1 Détection de BLOB

Un BLOB, acronyme de Binary Large Object est une zone de pixels homogènes avec une intensité lumineuse comprise entre des niveaux prédéfinis. Cet outil permet la détection d'objets ou de surfaces ayant la même image Blob, en contrôlant la surface ou en comptant le nombre d'objets. Les systèmes de détection de BLOB permettent d'extraire d'une image des régions aussi claire ou obscure que l'environnement. Il y a deux classes de méthodes de détection de BLOB, celles basées sur les expressions dérivées et celles reposant sur les extrema locaux d'intensité du paysage.

L'étude de ces BLOB est complémentaire de celle des coins. En effet une image est, pour simplifier, un ensemble de contours, de coins et de BLOB. Au début de l'intérêt pour les BLOB, ceux-ci n'étaient utilisés que pour l'extraction de région en vue d'un traitement futur. Ces régions peuvent signaler la présence d'un objet ou d'une partie d'un objet et donc ont été utilisées pour la reconnaissance d'objets. Puis, petit à petit, l'idée de l'utiliser pour segmenter les images, et pour analyser les textures a émergé. Ce n'est que très récemment que les descripteurs de BLOB ont trouvé un engouement populaire dans le domaine de l'extraction des points d'intérêt [PER03]. La plus connue et utilisée des diverses méthodes pour la détection et l'extraction de BLOB est basée sur le calcul d'un Laplacien après l'application d'un filtre Gaussien. La détection de BLOB est un domaine toujours ouvert, un des plus connus moteur d'indexation et de recherche d'image par le contenu se base sur cette approche. BlobWorld a été développé par l'université de Berkeley et utilise la segmentation en régions pour indexer les images [BLO99].

2.5.2 Détection des coins

Un coin peut être défini comme l'intersection d'au moins deux traits. Un coin peut être aussi décrit comme un point pour lequel il y a deux différentes et dominantes directions de bordure dans le voisinage local du point. Malheureusement, dans la littérature, les notions de primitives structurelles, point d'intérêt et de coins se mélangent. On peut

souvent trouver un terme pour un autre et de plus, il n'est pas rare de voir intégrer dans ces méthodes les détections de blob et d'arête qui, si elles sont des méthodes philosophiquement proches donnent des traitements subséquents différents. La détection de coins a la réputation d'être peu robuste et nécessite souvent d'être supervisée par des experts ou encore d'une surabondance pour éviter l'effet néfaste des erreurs de détection sur l'indexation ultérieure.

Une méthode simple pour trouver les coins est d'utiliser une fenêtre de corrélation. Cette méthode est fastidieuse et coûteuse en temps, une façon un peu plus évoluée de réaliser la détection est d'utiliser l'algorithme de Moravec [MOR80].

L'algorithme de Moravec

Cette méthode consiste à considérer le voisinage d'un pixel et à déterminer les changements moyens d'intensité dans ce voisinage. Ceci couplé avec l'utilisation d'une fenêtre glissante adaptée. Plus précisément on considèrera la fonction E moyenne du changement d'intensité dans une fenêtre centrée.

$$E(x, y) = \sum_{u,v} w(u, v) |I(x + u, y + v) - I(x, y)|^2 \quad (251)$$

- w est la fonction caractéristique de la fenêtre considérée
- I est l'intensité du pixel u, v

A l'aide de la valeur de cette fonction on pourra déterminer dans quel cas de figure on se trouve.

- Ainsi, si la fonction E prend en (x, y) des valeurs faibles quelle que soit la direction du déplacement de la fenêtre, alors on considèrera que la fenêtre se trouve dans une zone homogène.
- Si la fonction ne prend des valeurs hautes que dans une direction, on considèrera que la fenêtre se trouve sur un contour.
- Enfin si la fonction prend des valeurs fortes pour toutes les directions de son déplacement alors nous sommes en présence d'un coin.

Pour résumer, le principe de la méthode est de chercher les maxima locaux de valeur minimale de E pour chaque pixel, tout ceci en déterminant un seuil.

Cette méthode souffre de nombreux défauts et a été introduite ici uniquement car elle sert de point de départ à une méthode beaucoup plus populaire qui corrige les imperfections et est connue sous le nom de méthode de Harris [HAR88].

- Le détecteur de Harris et Stephen

Harris et Stephen sont partis des imperfections de la méthode de Moravec et ont donné des méthodes pour les corriger.

La méthode est dépendante de la direction (anisotropique), en effet pour ne pas surcharger le temps de calcul, la fenêtre ne se déplace pas dans toutes les directions mais adopte un pas de 45 degrés. Pour améliorer cette limitation, on a recours à l'utilisation d'un développement de Taylor permettant l'approximation de la fonction d'intensité I plusieurs fois dérivable au voisinage d'un point, par une fonction polynôme dont les coefficients dépendent uniquement de la valeur des dérivées de la fonction en ce point.

Ainsi :

$$I(x+u, y+v) = I(u, v) + x \frac{\partial I}{\partial x}(u, v) + y \frac{\partial I}{\partial y}(u, v) + o((x^2, y^2)) \quad (252)$$

D'où :

$$E(x, y) = \sum_{u,v} w(u, v) \left| x \frac{\partial I}{\partial x}(u, v) + y \frac{\partial I}{\partial y}(u, v) + o(x^2, y^2) \right|^2 \quad (253)$$

Comme $o((x^2, y^2))$ est négligeable pour les faibles déplacements on considère finalement :

$$E(x, y) = Ax^2 + Cxy + By^2$$

Avec

$$\begin{aligned} A &= \frac{\partial I^2}{\partial x} \otimes w \\ B &= \frac{\partial I^2}{\partial y} \otimes w \\ C &= \left(\frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \right) \otimes w \end{aligned} \quad (254)$$

Une autre limitation est que la réponse de Moravec est bruitée car le filtre utilisé, w est de nature binaire 0 ou 1 en fonction de la position soit en dehors ou en dedans de la fenêtre. Pour pallier ce problème, Harris décide d'utiliser un filtre Gaussien permettant d'appliquer un léger flou sur le voisinage. La fonction Gaussienne utilisée est :

$$G(u, v) = \exp\left(-\frac{u^2 + v^2}{2\sigma^2}\right) \quad (255)$$

Un dernier inconvénient pallié par la méthode de Harris est le fait que le détecteur de Moravec répond de manière forte aux contours car pour chaque pixel on ne prend en compte que la valeur minimale de E. Harris décide donc de réécrire la fonction à l'aide d'une matrice.

$$E(x, y) = (x, y) \cdot M \cdot (x, y)^t$$

Avec :

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad (256)$$

Le comportement de la fonction E est caractérisé par la matrice M dont les valeurs sont les courbures de la fonction E.

Ainsi pour détecter les coins, Harris et Stephen décrivent un opérateur R de formule :

$$R = \text{Det}(M) - k\text{Trace}(M)^2$$

Avec :

$$\text{Det}(M) = AB - C^2 \quad \text{et} \quad \text{Trace}(M) = A + B \quad (257)$$

- Si R est positif alors c'est que l'on se trouve au voisinage d'un coin.
- Si R est négatif c'est que l'on se trouve sur un contour.
- Et si R est faible alors c'est que l'on se trouve dans une zone d'intensité constante.

Les méthodes de détection des coins se révèlent efficaces dans une certaine mesure, en effet ces méthodes ne sont pas robustes à une trop grande déformation de l'image et deux photos d'une même scène, prises avec deux angles différents peuvent se révéler très éloignées l'une de l'autre en terme de similarité visuelle. Certaines primitives intrinsèques à l'image sont peu voire pas du tout utilisées comme les couleurs ou les textures et seule la structure de l'image importe. C'est pourquoi nous nous proposons d'étudier dans la section suivante les méthodes s'attachant plus à l'image dans sa globalité.

2.6 Approches statistiques

Ces approches vont se charger d'extraire des primitives calculées sur toute l'image.

Contrairement aux approches structurelles ou locales qui s'intéressent à caractériser indépendamment différentes parties d'une image, les méthodes globales vont, à l'aide de statistiques décrire l'image comme un tout. Ces méthodes ont été souvent critiquées car elles semblent de prime abord moins complexes ou "intelligentes" que les méthodes structurelles. En revanche leur simplicité d'application leur donne un gain de rapidité notable en termes de temps d'exécution.

De plus, les études en cognition semblent mettre en avant que notre cerveau aurait un comportement global dans le traitement des informations visuelles. Par exemple, il semblerait que les visages soient mieux reconnus quand ils sont représentés de trois quarts gauche que de trois quarts droit [FIO94b] [FIO96] [FIO97]. Des parties différentes de notre cerveau s'activent quand nous devons reconnaître tel ou tel type d'images. Par exemple pour les visages, il semble que ce soient les lobes temporaux qui soient en pleine effervescence lors de leur détection. Cette activation se fait en amont, c'est-à-dire que les parties du cerveau s'activent avant de voir l'image, car on sait ce que l'on s'apprête à chercher. Il semblerait aussi que notre cerveau dispose d'un panel de méthodes pour reconnaître les objets et que celles-ci soient stockées dans différentes parties de notre tête. Ce qui ressort, c'est que, toujours en reconnaissance visuelle, nous avons une analyse en deux phases et la première serait globale.

C'est pourquoi nous nous attacherons dans cette section à présenter les méthodes dites globales les plus représentatives. Nous commencerons par la méthode la plus utilisée dans l'indexation d'image et sûrement la plus ancienne qui est celle de l'analyse de l'histogramme de couleur d'une image. Nous verrons aussi les méthodes issues de l'analyse des histogrammes. Puis nous nous intéresserons ensuite aux champs de Markov et à leur application dans le domaine de l'indexation d'images pour finalement finir par survoler les méthodes dites d'extraction d'invariants.

2.6.1 Les histogrammes

L'histogramme d'une image est un élément statistique permettant de représenter la distribution des intensités des pixels de l'image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse. Par convention, un histogramme d'image représente le niveau d'intensité en abscisse en allant du plus foncé (à gauche) au plus clair (à droite), reposant ainsi sur la relation d'ordre qui existe sur les niveaux de gris.

Ainsi, l'histogramme d'une image en 256 niveaux de gris est représenté par un graphique possédant 256 valeurs en abscisse, et le nombre de pixels de l'image en ordonnée. Prenons par exemple l'image suivante en niveaux de gris :



Son histogramme est :

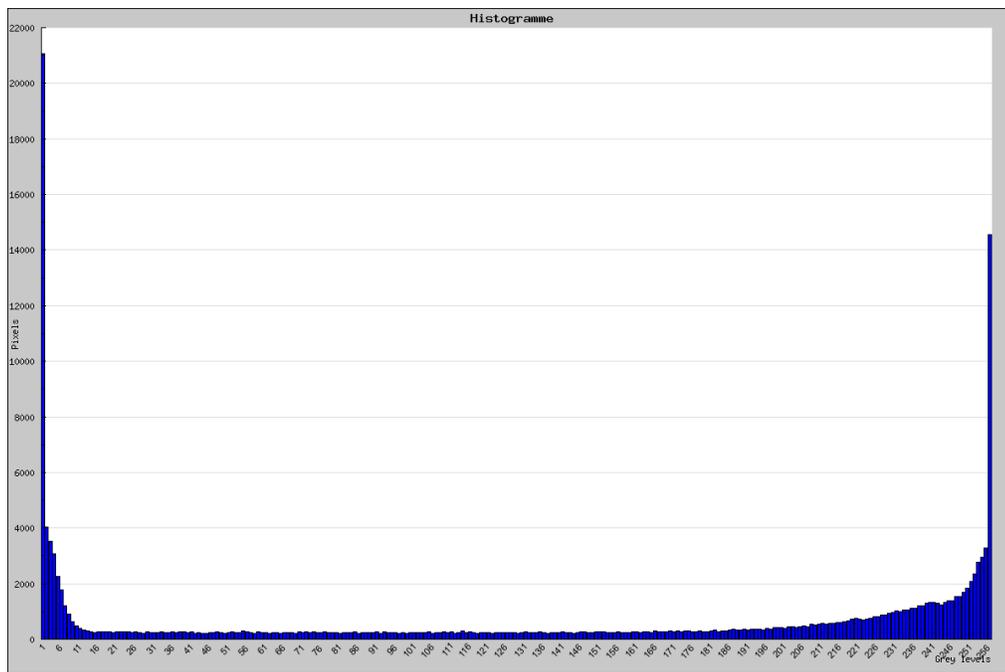


Fig 2.6.a : Exemple d'image et d'histogramme associé

Cet histogramme montre que les couleurs noires et blanches sont les plus présentes dans cette image.

L'histogramme d'une image a servi au début à traiter l'image en vue de par exemple harmoniser la répartition des niveaux de luminosité, ou encore augmenter son contraste en étirant l'histogramme sur l'ensemble des valeurs disponibles. Ainsi sur une image,

cette opération aura pour résultat de rendre les pixels clairs encore plus clairs et les pixels sombres encore plus sombres.

L'idée est venue plus tard d'utiliser cet histogramme pour indexer une image. En effet celui-ci renferme des informations importantes sur la répartition colorimétrique des pixels. Ce sont les travaux de M.J. Swain et D.H. Ballard qui ont défriché le domaine de l'indexation par les histogrammes, ils ont été les premiers à affirmer qu'un objet ou une image était décrit par un histogramme de couleur [SWA91]. Leurs travaux ont poussé l'exploration du domaine à trouver des espaces de représentation de couleur pouvant rendre la comparaison invariante à l'éclairage.

De nombreuses méthodes pour comparer les histogrammes ont été développées, en 1997 Bernt Schiele [SCH97] a testé toutes les méthodes de comparaison d'histogrammes multidimensionnels d'image.

- Comparaison d'histogrammes par fonction d'intersection

C'est par cette fonction que Swain et Ballard ont réalisé leur comparaison d'histogrammes. Cette méthode comporte quelques limites dont celle qui suppose que toutes les mesures soient équiprobables. L'intersection de deux histogrammes H_1 et H_2 de deux images en niveaux de gris se formule ainsi :

$$\bigcap(H_1, H_2) = \sum_i \min(H_1(i), H_2(i)) \quad (261)$$

Les avantages de cette formule est qu'elle est très rapide à calculer, en revanche elle marche bien pour les histogrammes ayant un grand nombre de classes et serait peu efficace sur des histogrammes en niveaux de gris non uniformément répartis.

- Comparaison d'histogrammes par distance quadratique

Les distances quadratiques entre deux histogrammes peuvent être écrites en utilisant une matrice carrée de poids W . La matrice utilise les relations entre les différentes classes de l'histogramme.

$$Quad(H_1, H_2) = (Q - V)' W (Q - V) \quad (262)$$

Une matrice spéciale donne la distance de Mahalanobis la distance devient :

$$Maha(H_1, H_2) = \sum_i \frac{(h_1 i - h_2 i)^2}{\eta_i^2} \quad (263)$$

Avec η_i^2 la variance de la classe i . En fait cela revient à choisir comme matrice W l'inverse de la matrice de covariance des valeurs de l'histogramme.

- Le test du χ^2

Le test du χ^2 permet de montrer que deux ensembles ne sont pas extraits de la même population. Ainsi, si l'on n'arrive pas à prendre cette décision, c'est que les ensembles viennent de la même population.

Pour comparer deux histogrammes on en considèrera un comme étant un modèle car le test du χ^2 doit normalement comparer une population avec un modèle. La formule sera :

$$\chi^2(H_1, H_2) = \sum_i \frac{(h_{1i} - h_{2i})^2}{h_{1i} + h_{2i}} \quad (264)$$

Même si aujourd'hui les intersections histogrammes restent la signature la plus courante pour comparer des images, il n'en reste pas moins que certains problèmes persistent. Le premier est la taille de ceux-ci, trop importante pour permettre une indexation rapide. Couplé à la complexité des méthodes de comparaison d'histogrammes, ceci nous interdit de réaliser la recherche à la volée. Il faut, comme dans la plupart des moteurs de recherche d'image par le contenu, pré indexer les images de la base et garder ces index pour n'avoir, au moment de la recherche qu'à traiter l'image requête. Le problème qui est soulevé, c'est que les valeurs qui représentent une image sont nombreuses. Elles vont de 256 pour une image en niveaux de gris, jusqu'à 256^3 (16777216) pour un histogramme d'une image en RGB. On va devoir indexer les histogrammes par une simple table de hachage pour accélérer la recherche. Un exemple est l'utilisation de moments d'histogramme comme la moyenne [HAF95].

La deuxième limitation est leur sensibilité aux faibles variations de luminosité, ce qui rend difficile la comparaison d'images similaires, acquises dans des conditions différentes. La réponse à ceci a été, pour certains, de changer d'espace colorimétrique. Ainsi l'espace RGB, espace standard en traitement d'image, traîne avec lui un problème non négligeable, en effet les trois composantes Rouge, Verte et Bleue restent corrélées. Ainsi, difficile de traiter des images dont l'histogramme dépasse la dimension un. Une réponse a été de passer vers d'autres espaces comme l'espace TSV qui permet de caractériser une couleur en fonction de sa teinte T qu'on pourrait qualifier de type de couleur (jaune, vert, marron ...) et qui prend une valeur comprise entre 0 et 360, sa saturation S que l'on pourrait nommer l'intensité de la couleur qui varie de 0 à 100% et sa valeur V ou brillance qui varie aussi de 0 à 100%. Ainsi en utilisant, cet espace de couleur, il devient possible de choisir ce que l'on veut indexer dans l'image, ses couleurs indépendamment de la luminosité ou encore l'intensité des couleurs indépendamment de la teinte elle-même.

Le troisième problème posé par l'utilisation des histogrammes est que ces méthodes font complètement abstraction de la répartition des couleurs au sein de l'image. Ainsi, il a été constaté que des images totalement différentes peuvent avoir des histogrammes identiques ou presque. Une réponse à ce problème a été de segmenter l'image avant d'extraire les histogrammes. Ces méthodes donnent un caractère géographique plus prononcé aux histogrammes mais décuplent la limitation introduite plus haut du nombre de classes des histogrammes.

2.6.2 Les moments statistiques et histogrammes cumulés

L'utilisation des histogrammes conserve un problème important qui est le trop grand nombre de primitives à sauvegarder. De plus, la difficulté qu'apporte la comparaison d'histogrammes quand ceux-ci sont décalés l'un par rapport à l'autre ou semblables mais pas identiques a amené certains chercheurs à essayer de pallier ces problèmes. Ainsi Markus Stricker et Markus Orengo ont proposé deux méthodes utilisant des histogrammes cumulés ainsi que les moments des histogrammes pour indexer une image [STR95]. Dans cette étude il est proposé d'utiliser les histogrammes cumulés d'une image pour l'indexer. Dans un histogramme cumulé, la valeur associée à chaque classe est le nombre de pixels ayant le niveau concerné par la classe ou un des niveaux inférieurs. La robustesse de cette approche permet de travailler dans des espaces de couleur grossièrement quantifiée.

La deuxième méthode est de ne stocker que les trois premiers moments statistiques de chaque canal d'une image comme primitives.

Les trois premières primitives sont les valeurs moyennes de couleur d'un canal. Les deux autres du groupe de trois primitives sont l'écart type et l'aplatissement des trois canaux.

Ainsi on aura pour chaque canal de couleur :

$$E_i = \frac{1}{N} \sum_{j=1}^N p_{ij} \quad \text{et} \quad \sigma_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^2 \right)^{\frac{1}{2}} \quad \text{et} \quad s_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^3 \right)^{\frac{1}{3}} \quad (265)$$

p_{ij} est la valeur du $j^{\text{ème}}$ pixel de l'image dans le $i^{\text{ème}}$ canal.

Comme les primitives sont de même unité nous pouvons déduire que la distance entre deux histogrammes H et H' de deux images se calcule ainsi :

$$d(H, H') = \sum_{i=1}^r \omega_{i1} |E_i - E'_i| + \omega_{i2} |\sigma_i - \sigma'_i| + \omega_{i3} |s_i - s'_i| \quad (266)$$

Avec r le nombre de canaux de couleur, et les ω_{il} qui sont des poids donnés par

l'utilisateur. Les tests semblent montrer que cette méthode est plus robuste que les méthodes utilisant l'ensemble des valeurs de l'histogramme, mais le nombre de paramètres à fixer est important.

2.6.3 Les champs de Markov

Les champs de Markov sont largement utilisés en analyse d'image [AZE87] [DUB89]. Ils font partie des techniques de base de la discipline. Aux débuts, utilisés dans le domaine de la physique statistique, ils ont été appliqués aux réseaux à deux dimensions qui peuvent modéliser les images. La définition d'un champ de Markov s'énonce ainsi :

X est un champ de Markov si et seulement si la probabilité conditionnelle en un pixel n'est fonction que de la configuration du voisinage du pixel considéré.

Soit :

$$P(X_8 | X_t, t \neq p) = P(X_8 | X_t, t \in V_8) \quad (267)$$

Avec p le pixel considéré et V_8 le voisinage du pixel.

Ainsi, le niveau de couleur ou de gris d'un pixel ne dépend que de celui de ses voisins. Cette hypothèse se justifie bien dans un grand nombre d'images de synthèse mais aussi dans les images naturelles souvent formées de zones homogènes en textures. C'est d'ailleurs en analyse de texture que les champs de Markov sont largement utilisés [STO98]. Ils sont très efficaces dans la détection de zones de même texture. Les champs de Markov permettent de modéliser des propriétés globales en utilisant des contraintes locales dans les images.

En revanche les champs de Markov sont très gourmands en temps de calcul et ils doivent être finement paramétrés car, en fonction de l'échelle d'observation, on peut affirmer que n'importe quelle image n'est qu'une seule et même texture.

2.6.4 Les invariants

Ces approches traduisent le problème central de l'indexation d'image, le choix de primitives pertinentes traduisant de façon fiable le contenu sémantique d'une image. Elles posent de plus une hypothèse, c'est qu'il n'y a pas de solution générale pour extraire des primitives pertinentes d'une image et ceci parce qu'il n'existe pas d'attribut modélisant une base d'images selon tous les points de vue utilisateur. On peut donc chercher à extraire des descripteurs en fonction du contexte. Un invariant est un élément d'un ensemble qui est sa propre image dans une transformation ponctuelle. Dans le domaine de l'image cela se traduit par l'extraction de primitives qui restent les mêmes si une image subit une transformation quelconque comme une déformation ou un

changement d'illumination.

Plusieurs classes d'extraction d'invariants peuvent être considérées. La première classe contient les invariants calculés algébriquement, obtenus à partir de quotients ou de puissances de moments. On trouve dans cette catégorie par exemple les moments de Hu ou encore les moments de Zernike [TEA80-CHO03].

La deuxième catégorie se compose des méthodes se basant sur des transformations d'image comme la transformée de Fourier ou celle de Fourier-Mellin [DER00-ADA01].

La troisième classe est celles des méthodes qui utilisent des réseaux de neurones [FAU06]. En effet, cela permet souvent de rendre invariant aux déformations géométriques la reconnaissance d'un objet. Pour cela on présente, pour l'apprentissage par réseau de neurones différents échantillons de l'objet à différentes échelles, orientations...

Une quatrième classe est celle des méthodes qui sont une combinaison de différentes méthodes explicitées ci-dessus.

Les méthodes d'indexation par invariants stipulent de connaître au préalable à quoi on doit être invariant et sur quel type d'images on travaille. Cette hypothèse interdit de pouvoir trouver des invariants, voire même une catégorie d'invariants qui resteraient valables quel que soit le type de transformations ou le type d'images auxquels on serait confronté.

2.6.5 Bilan

La méthode par histogramme est simple et efficace. En revanche elle perd toute notion de structure au sein de l'image. Les autres méthodes présentent de nombreux paramètres ce qui les rend plus difficile à mettre en œuvre.

Nous avons donc tourné nos recherches vers une méthode globale simple permettant la modélisation d'une distribution. C'est pourquoi nous explorons, dans le chapitre suivant, le domaine des lois puissances.

LES LOIS PUISSANCE

3 Les lois puissance

Dans toutes les méthodes que nous venons de passer en revue, nous avons vu que l'essentiel de la recherche s'exerce au niveau de la recherche de descripteurs, qu'ils soient globaux ou locaux, qu'ils traduisent une texture ou une représentation structurelle à partir d'une segmentation de l'image. On peut remarquer, au niveau des résultats obtenus par les systèmes actuels d'indexation, que la simplicité de l'histogramme donne des résultats de qualité étonnante par rapport à ceux obtenus par des méthodes beaucoup plus sophistiquées et de plus haut niveau cherchant à mettre en évidence les différents éléments du contenu de l'image. Il nous a semblé intéressant de rechercher dans une méthode statistique globale si l'on ne pouvait pas mettre en évidence au niveau du pixel des relations vérifiées par la totalité des informations captées au niveau du pixel ou perçues sur chaque pixel.

Les relations les plus simples sont les relations linéaires et il semblerait qu'aucune relation de ce type n'apparaisse de manière évidente dans une image, excepté lors de la détection de droites, que celles-ci soient au niveau d'un contour ou représentant un trait. Les modèles plus complexes sont rarement mis en œuvre. La nature est pourtant essentiellement non linéaire, et les lois gaussiennes ont montré tout leur intérêt d'un point de vue théorique aussi bien que dans les applications.

Dans cette section, après nous être attaché à décrire le fonctionnement et l'intérêt des lois puissance, nous en étudierons quelques unes celles qui semblent aujourd'hui les plus représentatives. Nous avons sélectionné la loi de Pareto, la loi de Bendford, les relations qui interviennent dans le domaine des fractales et enfin les deux lois de Zipf sur lesquelles repose notre travail.

3.1 Définition

Le terme "loi puissance" désigne un modèle de loi qui reflète une relation vérifiée entre deux variables que nous supposerons par la suite, scalaires. Cette relation entre les variables x et y est de la forme suivante :

$$y = ax^k \quad (311)$$

Où a et k sont des paramètres qui permettent d'ajuster la loi.

La représentation graphique associée, dans un repère orthogonal, est illustrée dans la figure ci-après

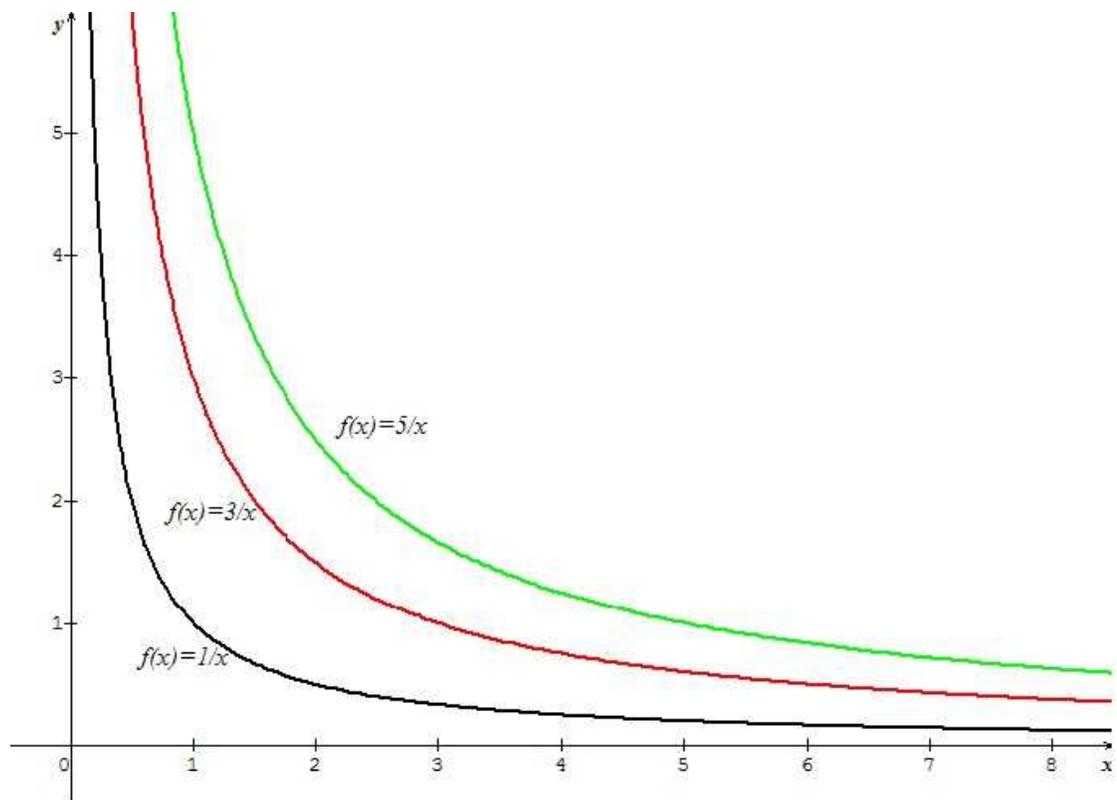


Fig 3.1.a : Exemple de courbes associées à des lois puissance

Une loi puissance n'est évidemment pas linéaire mais une de ses propriétés intéressantes réside dans le fait que l'on peut facilement se ramener à une relation linéaire par un changement de variables. Ainsi les propriétés connues dans le monde du linéaire pourront être utilisées pour simplifier les calculs. Dans un repère dont les échelles sont logarithmiques, le graphe représentatif de la relation exprimant une variable en fonction de l'autre est une droite dont k est le coefficient directeur. Cette nouvelle relation

peut s'énoncer ainsi :

$$\log(y) = k \log(x) + \log(a) \quad (312)$$

Cette nouvelle relation est celle d'une fonction affine ($Y = f(X) = k.X+b$) dont le graphe associé est une droite de pente k et d'ordonnée à l'origine le logarithme de la constante de proportionnalité a .

La représentation graphique d'une loi puissance dans un repère bi-logarithmique devient :

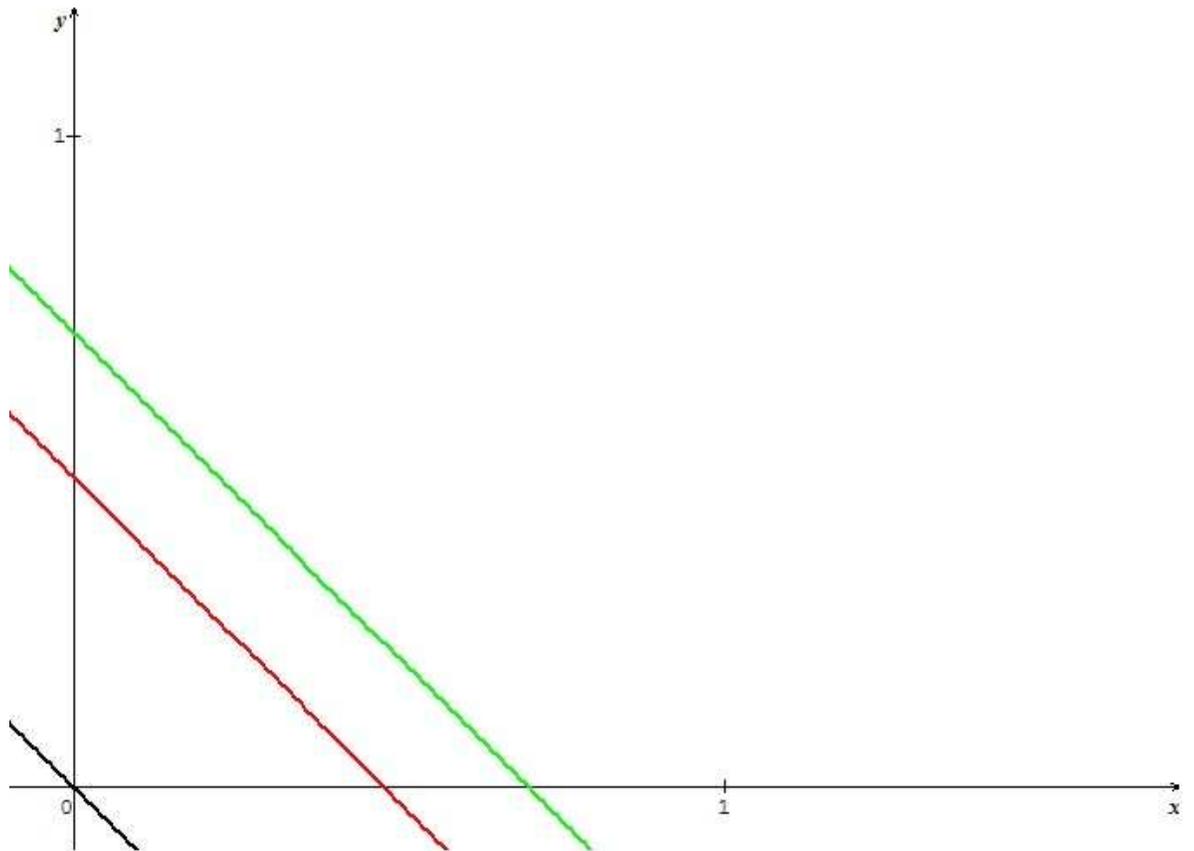


Fig 3.1.b : Courbes associées à des lois puissance dans un repère bi logarithmique

On remarque sur cette figure que la variation de la constante a n'entraîne aucune modification de l'aspect de la droite ce n'est que son ordonnée à l'origine qui change.

Si l'on fait varier maintenant k nous obtenons les courbes suivantes :

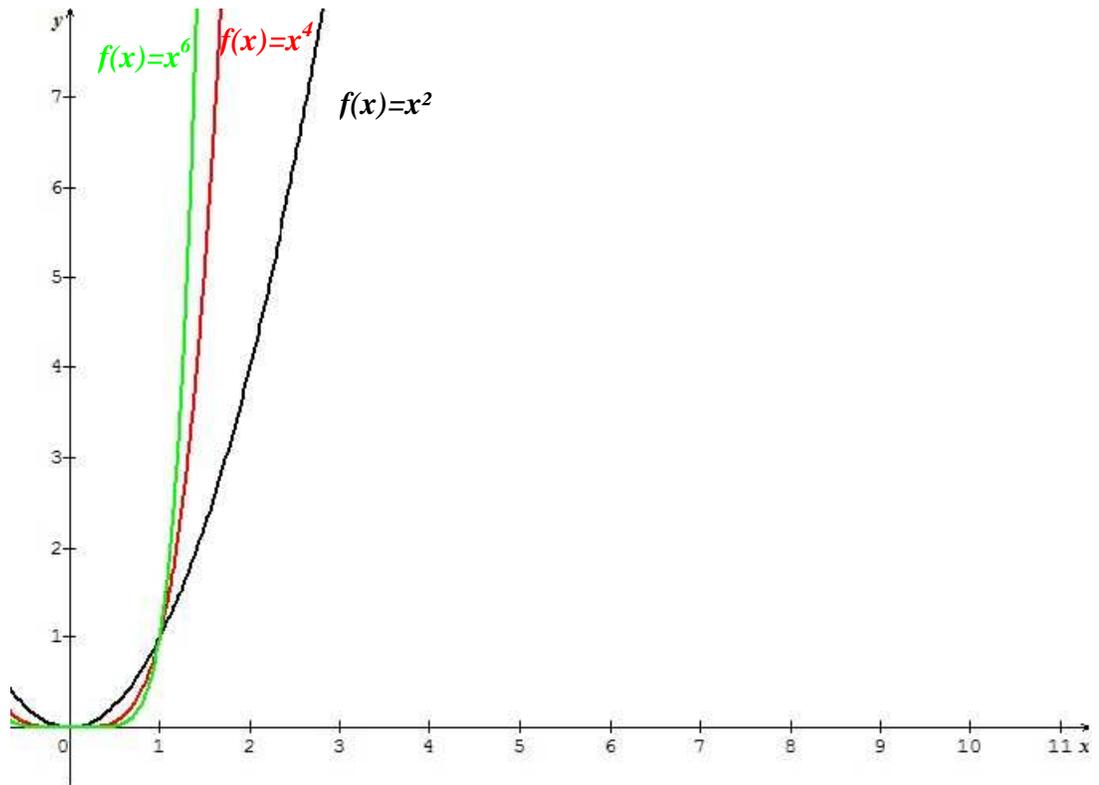


Fig 3.1.c : Exemple de courbes de lois puissance

Si on observe l'influence de la variation de k sur les courbes tracées dans un repère dont les échelles sont logarithmiques nous obtenons les graphes suivants :

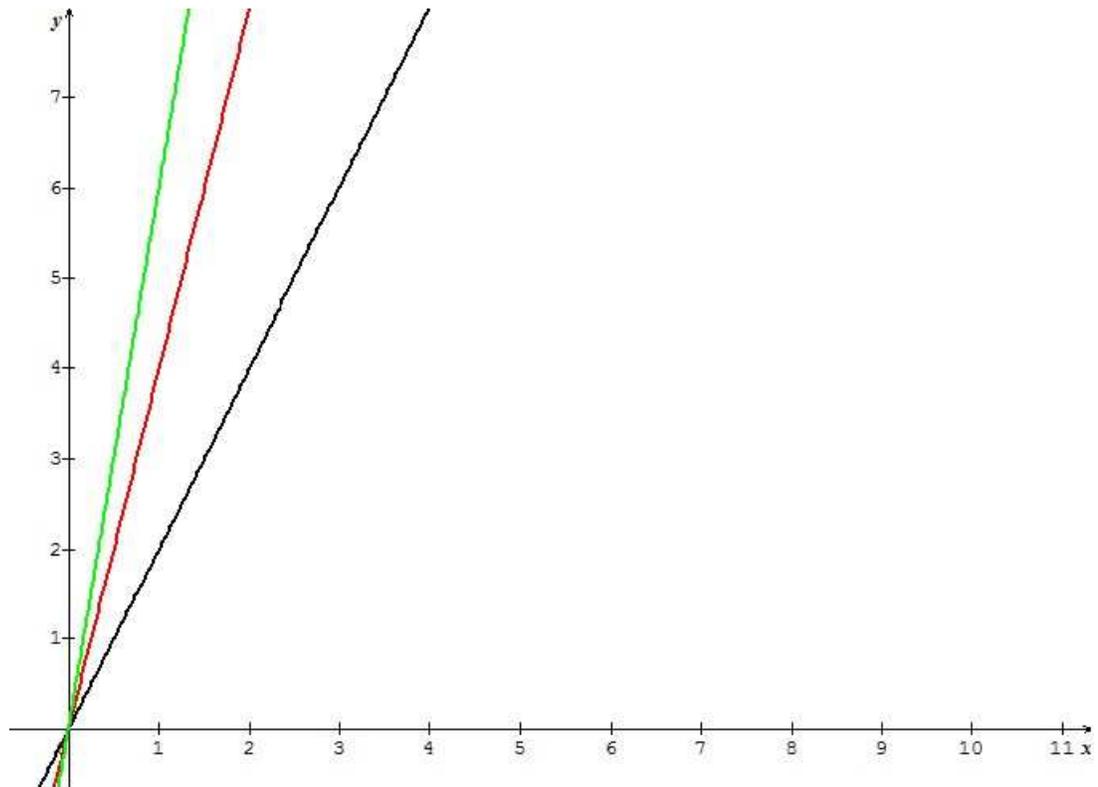


Fig 3.1.d : Courbes de lois puissance dans un repère bi logarithmique

3.2 Les principales lois puissances

Dans cette partie nous passerons en revue quelques lois puissance qui sont communément utilisées dans différents domaines d'étude. Il est à remarquer que ce sont des lois qui ont été empiriquement observées et ensuite énoncées puis reconnues universellement. Nous nous arrêterons en particulier sur la loi dite de Zipf sur laquelle repose notre approche.

3.2.1 La loi de Pareto

L'économiste italien Vilfredo Fr d rico Damaso surnomm  par ses  tudiants Marquis de Pareto est le premier   mettre en  vidence l'application d'une loi puissance   un ph nom ne  conomique. En 1906 il  tudie la r partition des revenus des habitants de divers pays industrialis s et constate que 80% des richesses appartiennent   20% de la population et ce, quel que soit le pays observ  [ALP84]. Cette observation donne naissance   une loi empirique  non ant que cette r partition suivrait une loi puissance.

L'énoncé de la loi a été fait dans un cadre probabiliste. Concrètement si l'on prend un individu au hasard dans la population, la probabilité que ses revenus associés à la variable aléatoire X , soient supérieurs à une somme x suit la relation suivante :

$$P(X > x) = \left(\frac{x}{x_{\min}}\right)^{-k} \text{ avec } x \geq x_{\min} \quad (321)$$

k est une constante et x_{\min} est le revenu minimum possible dans la population observée.

Il est à noter que l'espérance d'une variable aléatoire suivant la loi de Pareto est :

$$E(X) = \frac{kx_{\min}}{k-1} \text{ si } k \text{ est supérieure à } 1 \text{ et si } k \leq 1 \text{ alors } E[X] \text{ n'est pas définie sur } \mathbb{R}.$$

Ainsi k est comme dans toute loi puissance la pente de la distribution rapportée à un graphe logarithmique et modélise l'inégalité des revenus de la population.

La constante k est la pente de la représentation graphique de la distribution des revenus relatifs, observés dans la population, dans un repère bilogarithmique et elle modélise l'inégalité des revenus de la population.

La loi de Pareto et ses dérivées ont été appliquées surtout en gestion et de nombreuses études ont montré que l'on pouvait modéliser un grand nombre de phénomènes managériaux grâce à cette loi puissance. Par exemple :

- ✓ En gestion des stocks, sous sa forme de loi ABC de Joseph Juran qui énonce que les ressources représentant 70 à 80% du CA sont rassemblées dans la classe A, les ressources contribuant entre 10 à 15% au CA sont dans la classe C et la classe B rassemble les ressources intermédiaires. Juran déclare que :

Le principe de Pareto permet seulement de séparer les choses en deux parts. En réalité, il existe 3 parties. La troisième est un « résidu » qui prend place entre les composantes prioritaires et les composantes secondaires. Ce « résidu » peut être dénommé « zone à risques » (awkward-zone). Chaque élément de cette zone à risques n'est pas assez important pour justifier un lourd investissement dans l'analyse, mais leur regroupement dépasse les capacités d'analyse

- ✓ On peut aussi formuler que 20 % des produits représentent 80 % du chiffre d'affaires. Cela permet de déterminer sur quels processus on doit apporter des modifications en priorité.
- ✓ En gestion de projet, où la loi s'énonce par 80 % d'accomplissement d'une mise au point nécessite 20 % de l'effort.
- ✓ En évaluation des risques, lorsque 80% des articles restant sont ramenés à des

niveaux raisonnables, alors, la pérennité de l'entreprise est assurée.

- ✓ En gestion des services, 80 % des réclamations proviennent de 20 % des clients.

Cette loi est omniprésente dans la gestion des grandes entreprises mondiales.

Il est à noter que par son origine sociale cette loi a été reprise par certaines personnes à des fins politiques pour énoncer que les inégalités dans les populations sont un phénomène purement naturel.

3.2.2 La loi de Benford

Cette loi [BEN38], aussi appelée loi des nombres anormaux et édictée par Frank Benford vers 1938, s'appuie sur une observation faite quelques années auparavant par Simon Newcomb, un astronome américain, qui s'était aperçu que l'usure des premières pages des tables de logarithme était plus importante que celle des dernières. Il en conclura que l'utilisation des premières pages d'un livre est plus importante que celle du milieu et encore plus que celle de la fin. Frank Benford appliquera cette observation à différents phénomènes en étudiant des milliers de données sélectionnées dans des domaines aussi divers que les longueurs de plus de 300 fleuves, les recensements démographiques de plus de 3 000 régions, les masses atomiques des éléments chimiques, les cours de bourse, les constantes de la physique, les couvertures de journaux. Il constata, que le premier chiffre était un "1" près d'une fois sur trois dans cette masse de données et a donné son nom à la loi. On constate dans tous les cas que les données sont distribuées selon les valeurs indiquées dans le tableau 3.2.1 ci-dessous et illustrées dans le graphe de la figure 3.2.a :

première décimale d'un nombre (d)	1	2	3	4	5	6	7	8	9
fréquence d'apparition (f)	30,1%	17,6%	12,5%	9,7%	7,9%	6,7%	5,8%	5,1%	4,6%

Tab 3.2.1 : Tableau de répartition fréquentielle de la loi de Benford

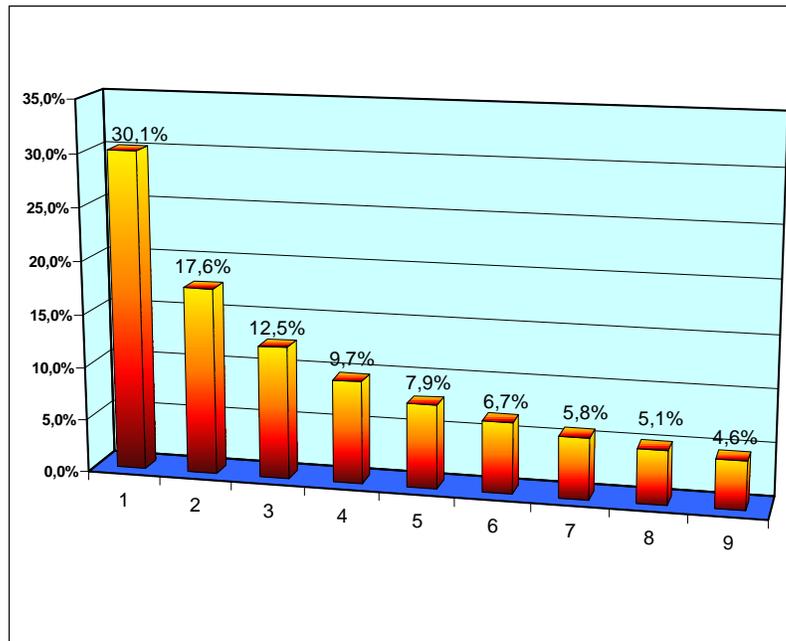


Fig 3.2.a : Histogramme de répartition fréquentielle de la loi de Benford

D'un point de vue plus formel, cette invariance rencontrée dans la nature peut être modélisée par une loi qui se résume dans la formule suivante :

$$f = \log_{10} \left(1 + \frac{1}{d} \right) \quad (322)$$

Ce n'est qu'en 1996 que Terence HILL démontra [HIL96] mathématiquement cette loi en généralisant la formule à toutes les bases :

$$f = \log_b \left(1 + \frac{1}{d} \right) \quad (323)$$

Avec b la base dans laquelle sont exprimés les nombres. Cette loi est actuellement utilisée entre autres pour détecter les fraudes fiscales. En effet il est extrêmement rare que des données falsifiées suivent la loi de Benford. Cette loi ne s'applique qu'aux données issues de mesures et est incapable par exemple de s'appliquer à des séries aléatoires, inutile d'essayer de trouver grâce à cette loi les futurs tirages du loto.

Cette loi a été testée et validée par Jean-Michel Jolion sur les images et en particulier sur les gradients de l'image [JOL01].

3.2.3 Fractales

La géométrie euclidienne est généralement reconnue pour bien modéliser le monde

dans lequel nous vivons. Elle englobe la notion d'espace vectoriel qui est associée aux transformations linéaires. Elle est adaptée à des objets assez réguliers. Par exemple, les surfaces sont supposées pouvoir être approximées en tout point de manière linéaire par un plan tangent. Ce n'est qu'assez récemment que des cas de courbes non modélisables ont été mis en évidence, par exemple par Weierstrass. Benoît Mandelbrot, en 1974, invente le terme de "fractale" [MAN77] en se basant étymologiquement sur le mot fractus qui en latin signifie cassé. Le monde des fractales est un monde bien plus vaste que le monde décrit par la géométrie euclidienne. Nous pouvons mentionner quelques constructions de courbes reconnues comme fractales :

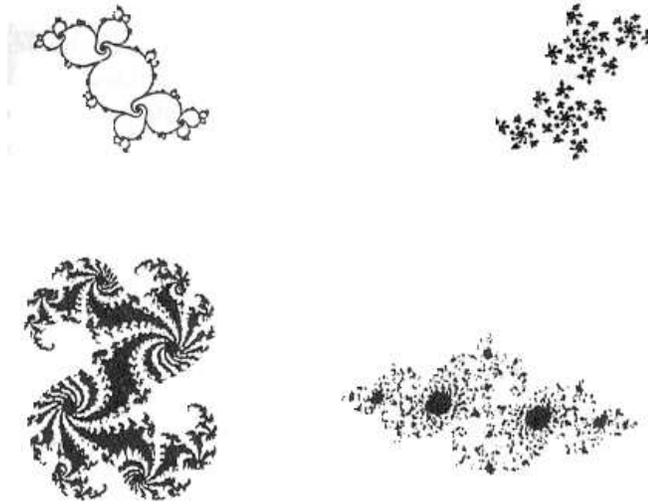


Fig 3.2.b exemples de courbes de Julia

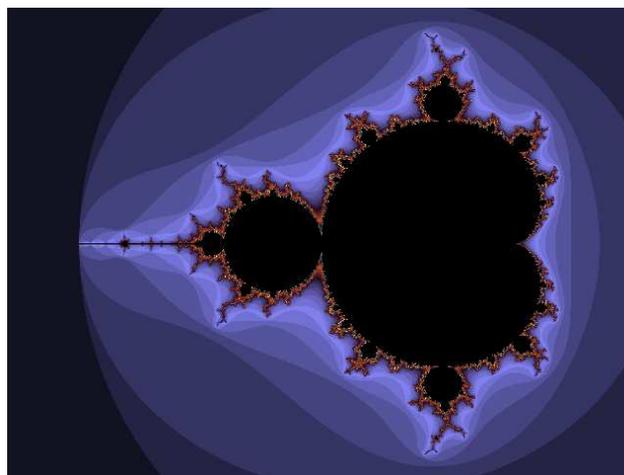


Fig 3.2.c ensemble de Mandelbrot

Mandelbrot a aussi introduit une notion quantitative pour qualifier la complexité des ensembles, c'est la dimension fractale. Il en existe de nombreuses définitions, prenons par exemple la dimension de boîtes.

$$P(\varepsilon) = \left(\frac{\varepsilon_0}{\varepsilon} \right)^D \quad (324)$$

D désigne la dimension du support de l'espace dans lequel est inscrit l'élément de dimension ε_0 .

Nous voyons que la notion de fractalité est fortement liée à l'existence d'une relation en loi puissance entre deux variables auxquelles on a accès par une mesure.

L'intérêt de l'introduction de ces modèles ayant recours à une loi puissance a été prouvé par les nombreuses applications qui ont utilisé cette approche dans des domaines très variés mais surtout, pour ne citer que celui qui se rapproche le plus de notre domaine, d'étude la compression d'images [JAC92].

Une fractale est une courbe ou une surface irrégulière qui se crée en suivant des règles déterministes ou aléatoires. D'abord utilisé sous forme d'adjectif il est maintenant couramment utilisé comme nom commun.

3.2.4 La loi de Zipf

Par ailleurs, c'est en 1949 un an avant sa mort que le linguiste George Kingsley Zipf a mis en évidence une nouvelle loi puissance. C'est en analysant l'œuvre de James Joyce "Ulysse" que Zipf se rendit compte en comptant les occurrences des mots dans celle-ci que le mot le plus courant revenait 8000 fois dans tout l'ouvrage, le dixième plus courant 800 fois, le centième 80 fois et le millième 8 fois. Il énonça donc une loi [ZIP49] qui, contrairement à la loi de Pareto qui repose sur des probabilités, utilise les fréquences et les rangs des individus, c'est-à-dire ici des mots. Selon Zipf, dans un ouvrage de langue anglaise les fréquences d'apparition des mots, ordonnées de la plus grande à la plus petite, respectent une distribution en loi puissance. Ainsi il met en évidence la relation suivante entre f la fréquence d'apparition d'un mot et r son rang relativement au texte étudié:

$$f = kr^a \quad (325)$$

Dans cette formule k et a sont deux constantes, la première étant positive et la seconde négative. La notion même de mot peut être revisitée, en effet des études ont montré qu'au-delà de mots définis comme l'unité élémentaire d'un texte, on peut considérer pour voir apparaître une distribution de Zipf des motifs. En effet, dans l'étude d'Ulysse on s'est

attaché à compter des ensembles de lettres ayant un sens mais il est aussi possible de compter des n-uplets de lettres avec n constant. Vérifier que la loi est respectée, permet de juger si un texte est suffisamment long pour être considéré comme représentatif d'un phénomène que l'on voudrait étudier.

Dans le cas des textes en langage naturel a est proche de 1 ce qui donne :

$$f \times r \cong \text{constante} \quad (326)$$

que l'on rencontre aussi sous la forme $f=k/r$

Ainsi pour la langue anglaise et son vocabulaire on obtient les fréquences suivantes :

Rang	Mot	Fréquence	Rang x Fréquence
1	The	69971	69971
2	Of	36411	72822
3	And	28852	86556
4	To	26149	104596
5	A	23237	116185
6	In	21341	128046
7	that	10595	76165

Tab 3.2.d : Répartition du vocabulaire anglais

Bien sûr la formule (326) n'est pas strictement vérifiée ici mais on n'a fait apparaître que les 7 mots les plus fréquents et nous verrons par la suite que ceux-ci ne sont pas toujours les plus représentatifs de la loi de Zipf. Il faut pour mieux appréhender la formule (326) regarder de façon complète et globale l'ensemble des mots et leur répartition.

Georges Zipf a énoncé que cette relation entre rang et fréquence est une propriété de tous les textes et que l'on peut tracer ce que l'on appellera dans toute la suite, "la courbe de Zipf" dans un repère avec une échelle bi logarithmique

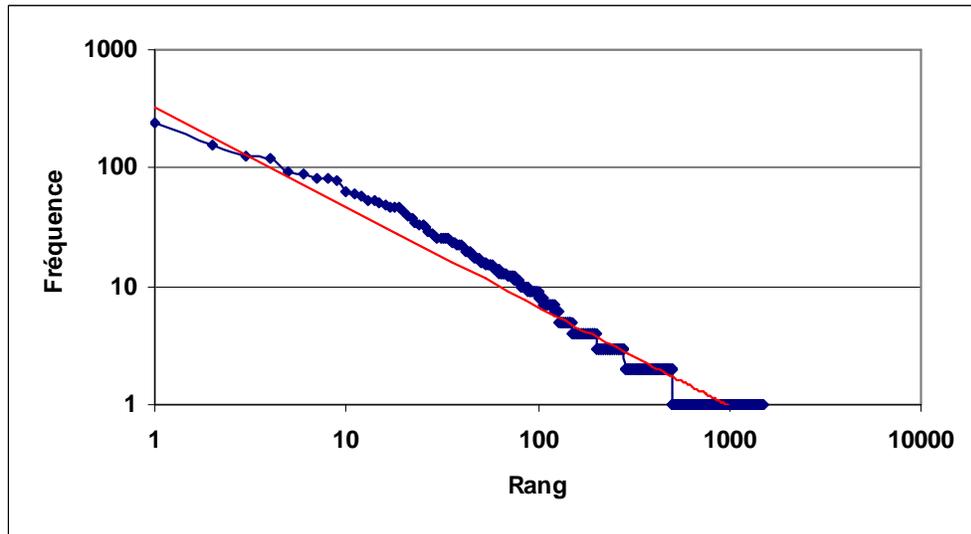
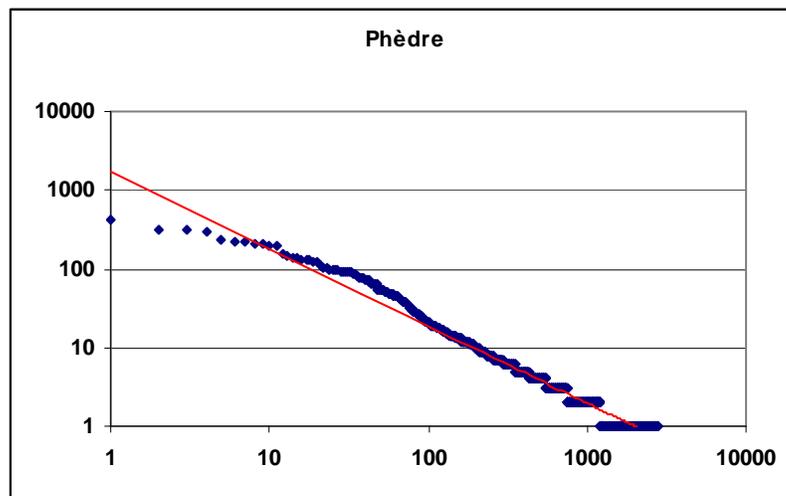


Fig 3.2.e : Courbe de Zipf caractéristique de la langue anglaise

Il a été vérifié que cette loi s'applique à de nombreux ouvrages. Nous en donnons quelques exemples dans les graphes qui suivent figure 3.2.e. On y a choisi les ouvrages de référence dans diverses époques.



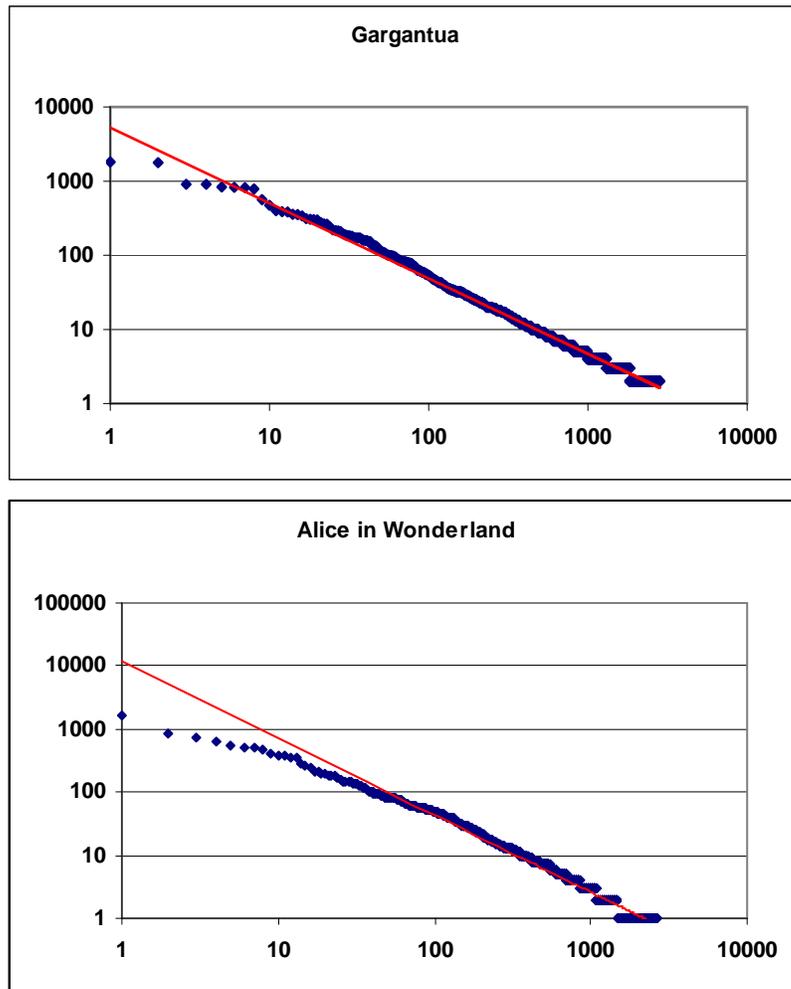


Fig 3.2.f : Exemples de courbes de Zipf de différents ouvrages

On constate que les graphes n'ont pas exactement la même forme, mais dans tous les cas on peut observer une grande linéarité des points en particulier environ après le dixième point. Il est à noter que les premiers points correspondent à des mots très fréquents qui ne sont pas vraiment porteurs de sens. Si l'on s'intéresse d'un peu plus près à ces courbes on pourra apercevoir que celles-ci sont composées en réalité de trois zones distinctes. La première zone observable est composée des premiers mots les plus fréquents et n'est pas en total linéarité avec les motifs suivants, on remarque en effet un changement de la pente de la droite de régression entre les points de début de courbe et ceux du milieu de courbe. Les mots de milieu de courbe eux respectent une linéarité presque parfaite et vérifient de ce fait leur distribution suivant une loi puissance. Les mots les moins fréquents donnent à la courbe un effet d'escalier caractéristique. Cet effet d'escalier est dû au fait que pour les basses fréquences il y a énormément de mots ayant

ces fréquences d'apparition. Cette observation déjà notée par différentes recherches sur la loi de Zipf [NEW05] nous amènera plus tard à considérer la courbe non plus comme un phénomène unique mais comme la juxtaposition de plusieurs phénomènes.

Beaucoup d'ouvrages dans des langues différentes se sont vus appliquer la loi de Zipf et cette loi a montré son efficacité. L'explication donnée par Georges Zipf a été de dire que le langage humain est un ensemble de règles très complexes à mettre en œuvre. Ainsi il y a un schéma particulier à mettre en place pour pouvoir créer une phrase bien formée. Pour résumer on peut affirmer qu'il est possible d'appliquer la loi de Zipf sur d'autres données du moment que l'on puisse percevoir une certaine structure. Cette constatation permet de ne plus cantonner la loi de Zipf aux textes mais à divers phénomènes monodimensionnels. Des chercheurs de l'Université de Californie ont mis en évidence que la répartition des plus grands tremblements de terre en Californie suivait la loi de Zipf, cette étude affirme pouvoir même prévoir la magnitude du prochain grand tremblement de terre grâce à une loi puissance [SOR96]. Il serait peut-être intéressant de vérifier si les dates des tremblements de terre suivent aussi une loi puissance.

Une autre application de la loi de Zipf fut sur la population des plus grandes villes de certains pays. On s'est rendu compte que les populations des plus grandes villes d'un pays suivaient une loi de Zipf [GAB99], le phénomène a été observé en premier au Brésil, puis au Maroc et dans d'autres pays. Une telle loi a même été vérifiée pour des régions peuplées récemment. Dans les pays jeunes, la population a tendance à se répartir dans les villes en fonction de l'importance de celles-ci.

Une étude a été aussi menée sur les utilisateurs du fournisseur d'accès AOL [ADA00] et sur le nombre de clics qu'ils réalisaient sur différents sites. Il y a une relation entre la fréquence des clics sur un site et son rang. AT&T a aussi réalisé une étude pour montrer que les appels téléphoniques ainsi que les courriels reçus par une personne suivaient la loi de Zipf [EBE02]. Beaucoup de phénomènes ont montré cette caractéristique, les explosions solaires [LUA91], les tailles de cratères sur la lune [NEU94], l'intensité des guerres [SMA82], l'état de santé des personnes les plus riches, la fréquence des noms de famille [MIY00], le nombre d'exemplaires vendus d'un ouvrage [HAC67].

En traitement du signal, la loi de Zipf a été vérifiée en analyse de signaux sonores [DEL03]. En effet on peut caractériser des signaux sonores et donc les indexer à l'aide de la loi de Zipf. La méthode reste la même et la difficulté est d'adapter le concept de mot ou de motif à des signaux sonores. Des primitives sont extraites des courbes de Zipf obtenues et permettent l'identification des signaux sonores.

Très récemment les physiciens Bernd Blasius, Ralf Tönjes et Andriy Bandrivskyy de l'Université de Potsdam en Allemagne viennent de montrer que les échecs suivent une

loi de Zipf [BLA07]. Les échecs présentent un nombre de parties possibles énorme. En revanche les débuts d'une partie, que l'on nomme ouvertures, sont relativement simples, de sorte que toutes les variantes ne sont pas explorées de façon uniforme. Ainsi les trois physiciens ont montré que le nombre de ces ouvertures jouées suivait une loi de Zipf. Ils ont mis en avant que la majorité des parties utilisent une minorité d'ouvertures.

Dans le domaine de l'image, la loi de Zipf a aussi été observée et utilisée, ainsi grâce à la loi on a pu mesurer la qualité de compression d'une image JPEG ou encore extraire les zones d'intérêt d'une image [CAR02].

3.2.5 La loi de Zipf inverse

On l'a vu, la loi de Zipf s'intéresse aux occurrences les plus fréquentes dans un phénomène naturel. La loi de Zipf inverse est une autre loi puissance dérivée de la première, mais qui pour sa part s'intéresse aux motifs les moins fréquents. En effet Zipf considère qu'aux motifs de fréquence faible on peut aussi associer une loi puissance. Elle peut s'énoncer par : le nombre de mots distincts $I(f)$ qui ont la fréquence f , est déterminé par la formule :

$$I(f) = l \times f^{-\beta} \quad (327)$$

Où l et β sont des constantes positives. Ainsi rapportée à un repère bi-logarithmique le graphe représentatif de la relation, la courbe de Zipf, prendra la forme d'une droite caractérisée par son coefficient directeur β . Pour le texte en langage naturel Zipf, estime β aux alentours de 2. Il peut paraître original pour représenter un phénomène de s'attacher aux motifs les moins représentés. Cette première mauvaise impression est mise à mal par les études menées sur la loi de Zipf inverse qui montre qu'elle est beaucoup plus discriminante que la loi de Zipf classique [COH97]. En effet sur le texte elle permet de mieux reconnaître un texte naturel d'un texte aléatoire. Elle semble donc totalement complémentaire de la loi de Zipf dite classique.

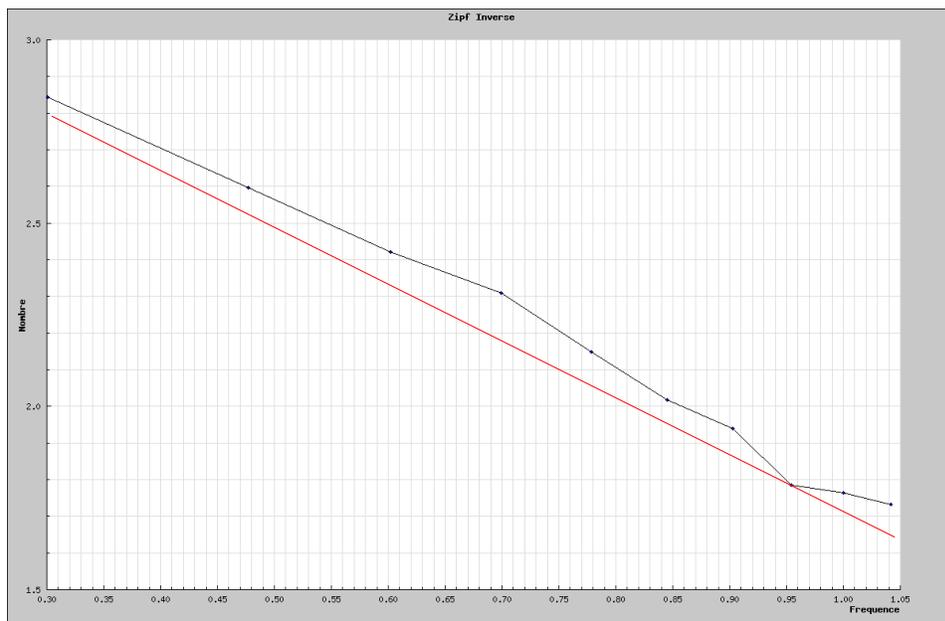


Fig 3.2.g : exemple de courbe de Zipf inverse tracée sur les dix motifs les moins fréquents

Si on observe la courbe de Zipf classique, on peut constater que pour les motifs les moins fréquents, ils sont souvent nombreux à avoir des fréquences d'apparition égales, ce qui donne un effet d'escalier à la courbe. Par ailleurs, dans certaines études sur la loi de Zipf [GUI02] les chercheurs montrent que les fréquences suivent de façon très conforme, la loi de Zipf mais une observation plus précise permet tout de même de constater trois parties dans la courbe. De gauche à droite, une première partie, en « zigzag », correspond aux motifs les plus fréquents puis une deuxième exceptionnellement linéaire correspond aux motifs moyennement fréquents puis en fin la partie en escalier.

La loi de Zipf inverse a été mise en avant dans de nombreuses études et s'est montrée plus efficace dans certains domaines que la loi de Zipf classique.

On a deux approches pour discriminer des phénomènes. D'une part on peut observer ce qui est le plus fréquent et donc apparaît évident à l'observation, mais alors on est confronté à une grande ressemblance entre des phénomènes assez proches. D'autre part, on peut observer des formes particulières qui ne sont pas utilisées très souvent mais qui sont très discriminantes, bien que n'entachant pas l'allure générale du phénomène étudié.

3.3 Application des lois puissances

Rapidement on s'est rendu compte que les lois puissance pouvaient modéliser un nombre important de phénomènes, soit naturels, soit sociaux, soit encore comportementaux. Que ce soit de la magnitude des séismes, à l'importance des avalanches en passant par les sites et blogs visités sur le web, toutes ces statistiques ont pu être modélisées en ayant recours à une loi puissance. Les lois puissance modélisent les phénomènes d'une façon globale et sont souvent adaptées à des volumes importants de données à traiter.

LA LOI DE ZIPF ET LES IMAGES

4 Le codage des motifs

La plus grande partie des études qui ont utilisé, ou plutôt vérifié, que la loi de Zipf était valide se sont bornées à des phénomènes monodimensionnels. Notre travail a pour objet l'étude et plus précisément l'indexation des images. Dans cette partie nous nous attacherons à décrire comment nous pouvons adapter le concept de motif utilisé dans les signaux monodimensionnels textuels, aux images. Nous décrivons plusieurs méthodes de quantification de l'alphabet qui décrit les nos pixels.

4.1 Le choix du motif

Pour adapter la loi de Zipf aux images, il nous faut transposer le concept de mot qui est un ensemble d'unités élémentaires pouvant prendre comme valeur un symbole de l'alphabet, au contexte des images. L'enjeu de cette phase est que la loi de Zipf ait statistiquement un sens. En effet, la loi de Zipf, dans le domaine monodimensionnel, ne peut s'appliquer que si le nombre de mots possibles dans un texte n'est ni trop faible, ni trop grand. Dans un cas comme dans l'autre, la courbe de Zipf n'aurait aucun sens. Si le nombre de mots possibles est faible ou immense, la courbe de Zipf phénomène à l'autre resterait une droite de pente nulle qui ne pourrait en aucun cas caractériser ces phénomènes. Un équilibre doit être trouvé entre un faible nombre de mots qui ne peuvent exprimer la diversité et un trop grand nombre qui rendrait peu maîtrisable statistiquement le cas particulier observé, équilibre qui semble trouvé quand est assurée la validité de la loi de Zipf. On peut penser que ce qui est vrai dans le choix de la taille d'un alphabet et d'un mot dans un univers monodimensionnel le reste dans des univers de dimension plus élevée, en particulier pour les images qui sont, elles, en deux dimensions.

La représentation de l'image que nous avons adoptée est un rectangle composé d'unités élémentaires nommées pixels. Notre alphabet sera composé des différentes valeurs que peuvent prendre ces pixels. Pour ce qui est des mots il nous faut trouver un ensemble de pixels qui donnera un sens aux pixels qui le constituent et d'où découlera le nombre de mots, que nous nommerons ici motifs, équilibré. La taille de notre alphabet, a priori au moins 256 valeurs, nous pousse à ne pas choisir un ensemble de pixels trop grand. Nous devons aussi tenir compte du fait que le nombre de pixels d'une image est faible devant le nombre de lettres dans un texte. Dans un texte, ce qui fait sens, c'est la succession de caractères, dans une image ce que l'on perçoit réellement c'est une zone, un bloc de pixels adjacents, un pixel isolé n'est pas significatif. Il semble donc le plus raisonnable de considérer comme mot un ensemble de pixels qui constituent un voisinage d'un pixel donné. Nous pouvons naturellement avoir recours aux voisinages 8-connexes ou aux voisinages 4-connexes qui sont régulièrement utilisés dans le traitement d'image [VIN93]. Les plus petits voisinages possibles comportent respectivement 9 et 5 pixels comme on peut le voir sur la figure 4.1.a. Ce sont eux qui constitueront le plus souvent les "motifs".

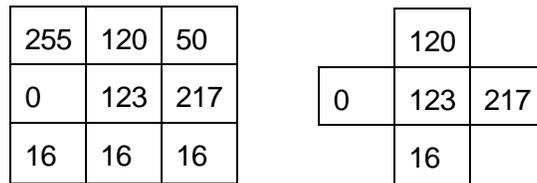


Fig 4.1.a : Exemples de motifs respectivement 8 et 4 connexes

Même en réduisant au maximum la taille des motifs, le nombre de motifs 8-connexes possibles reste très élevé, pour une image en niveaux de gris où chaque pixel peut prendre 256 valeurs différentes on obtient 256^9 motifs différents. Il faut comparer ce nombre au nombre des pixels d'une image.

En considérant ce procédé, on obtient des courbes de Zipf totalement horizontales, en effet la fréquence absolue d'apparition de chaque motif présent dans l'image est généralement proche de 0.

La forme du motif a certes un impact sur le nombre de motifs différents mais bien moins grand que l'impact de l'alphabet qui pour des images en niveaux de gris, comporte 256 symboles et 16777216 en couleur RVB.

Inutile d'expliquer que nous sommes loin du nombre de motifs possibles utilisés dans un alphabet de 26 lettres. Un constat s'impose, le nombre de pixels de la matrice du voisinage étant déjà réduit pratiquement à son minimum pour conserver les propriétés d'une topologie en dimension 2, il va falloir diminuer le nombre de niveaux de gris ou de

couleur ou de caractéristiques pris en compte pour un pixel d'une image.

4.2 La quantification des couleurs

Le travail sur la quantification des couleurs en niveaux de gris pour le codage de motifs a été réalisé dans la thèse d'Yves Caron sur la loi de Zipf et les images [CAR04]. Plusieurs méthodes ont été utilisées pour réduire le nombre de motifs nécessaires à l'extraction d'une courbe de Zipf significative sur des images en niveaux de gris. Nous revisiterons celles-ci, en proposerons de nouvelles et les utiliserons sur des images couleur. Cette étape est primordiale dans le mécanisme d'indexation car il aura un impact énorme sur les descripteurs que nous extrairons.

4.2.1 Les 9 classes

Cette méthode consiste à quantifier l'échelle de couleur de façon régulière et de façon indépendante du contenu d'une image. Par exemple si l'on souhaite réduire les 256 niveaux de gris à 9, on considère un histogramme de l'image en neuf classes dont les valeurs sont des intervalles de longueurs identiques et on attribue un symbole à chaque classe. On peut remarquer ici que s'il n'existe aucun ordre sur les lettres de l'alphabet hors de l'ordre lexicographique qui est tout à fait arbitraire, ici nous pouvons considérer une mesure de proximité entre les symboles qui a un sens par rapport à la perception visuelle humaine. On retrouve le manque d'une relation d'ordre naturelle entre les caractéristiques associées aux pixels quand on considère des images couleur.

En ce qui concerne ce codage, nous décidons de ne plus considérer 256 niveaux de gris possibles par pixel mais simplement 9. Si on reprend le calcul théorique, nous avons donc $(9)^{(3 \times 3)} = 387420489$ possibilités de motifs différents dans le langage. C'est un nombre encore très grand mais il n'est que théorique.

En effet, dans une image réelle, deux pixels côte à côte ont souvent des couleurs (NdG) proches l'une de l'autre. C'est pourquoi ce nombre reste très théorique et comme nous allons le voir, de nombreux motifs apparaissent définissant ce que l'on peut appeler des régions correspondant à une zone de couleur plus ou moins uniforme. C'est le plus simple d'entre tous les codages. Pour l'obtenir, il suffit de recalculer linéairement le niveau

de gris à traiter entre 1 et 9 grâce à la formule : $Y = X / \left(\frac{256}{3 \times 3} \right)$.

On obtient le tableau des correspondances indiquées dans le tableau 4.2.1.

0-27	28-55	56-83	84-111	112-139	140-167	168-195	196-223	224-255
1	2	3	4	5	6	7	8	9

Tab 4.2.1 : Classification de niveaux de gris en 9 classes

255	120	50	9	5	2
0	123	217	1	5	8
16	16	16	1	1	1

Fig 4.2.a : Exemple de motif avec le codage des 9 classes

Sur la figure 4.2.a on voit l'effet du codage sur un exemple de motif. Cette méthode n'est pas adaptée à une image particulière mais déduite des capacités du capteur, et ne tient pas compte de la distribution de l'histogramme de l'image étudiée. En effet, certaines classes seront plus ou moins peuplées en fonction du type d'image, certaines même seront complètement vides. Les inconvénients de la représentation par luminosité sont conservés.

4.2.2 Les rangs généraux

Le principe du codage par la méthode des rangs généraux est de numéroter les pixels composant un motif, entre 1 et 9 pour un voisinage 8-connexe. Les pixels sont rangés par ordre croissant du niveau de gris et le numéro est affecté en fonction du rang. Ainsi cette méthode permet d'avoir une information plus précise sur la texture de l'image au détriment de l'intensité. En analyse d'image, cette méthode a été utilisée pour les textures par de nombreux chercheurs [HAR85]. La méthode met en évidence les différences auxquelles l'œil est plus sensible qu'aux intensités elles-mêmes. La méthode retenue sera dérivée de celle de Duyan Bi [BI97]. La méthode semble bien adaptée à la détection de contours, en effet un contour se caractérise par une transition forte de niveau de gris entre pixels voisins, on base le codage sur une modélisation du changement des niveaux de gris dans une fenêtre. Notre méthode utilisera une fenêtre 8-connexe correspondant à la structure du motif énoncée plus haut.

Exemple 1 :

Motif rencontré :

241	222	180
32	158	112
58	58	59

Code du motif :

9	8	7
1	6	5
2	3	4

Fig 4.2.b : Exemple de motif avec le codage des rangs généraux

On peut noter que quelques problèmes peuvent être rencontrés. Par exemple, comment numéroté deux pixels de mêmes niveaux de gris ? Si on choisit de respecter l'ordre dans le motif alors nous introduisons une part aléatoire dans l'extraction de motif. Pour pallier ce problème on pourra décider de donner la même valeur à deux pixels ayant un niveau de gris identique.

Exemple 2 :

Motif rencontré :

241	222	180
32	158	112
58	58	59

Code du motif :

8	7	6
1	5	4
2	2	3

Fig 4.2.c : Exemple de motif avec le codage des rangs généraux

Cette méthode moins arbitraire révèle un défaut, elle introduit un surplus de motifs possibles. En effet dans la première méthode nous avons 9! motifs différents soit 362 880. Dans la deuxième le nombre de motifs possibles est de $4,1179 \times 10^8$ motifs différents. Le calcul du nombre de motifs différents possibles avec cette méthode est loin d'être trivial, en effet pour un K-voisinage le nombre de motifs pour un K-voisinage est :

$$N = \sum_{i=1}^K N_i(K) \quad (421)$$

$$\text{Où } N_1(K) = 1 \text{ et } N_i(K) = i^K - \sum_{j=1}^{i-1} \left(\binom{i}{j} \times N_j(K) \right) \text{ avec } (2 \leq i \leq K)$$

$$\text{Où } \binom{i}{j} = \frac{i!}{j!(i-j)!}$$

Pour plus de détails sur le nombre de motifs possibles on peut se référer au document [BI97].

Même si le nombre de motifs reste élevé il a beaucoup diminué par rapport au codage initial de l'image en niveaux de gris.

Un inconvénient majeur de la méthode est que pour des valeurs de niveaux de gris proches, la variation de rang peut être énorme.

Exemple 3 :

Motifs rencontrés :

0	28	56	11	12	14
80	107	134	15	16	17
165	200	235	18	20	21

Code du motif:

1	2	3
4	5	6
7	8	9

Fig 4.2.d : Exemple de motifs avec le même codage des rangs généraux

Visuellement les deux motifs indiqués dans l'exemple 3 de la figure 4.2.d donnent des perceptions visuelles très éloignées mais identiques quant au codage par la méthode des rangs généraux. On peut pour essayer de pallier, en partie ce problème, introduire un seuil pour tenir compte de l'amplitude de la variation des niveaux de gris à l'intérieur du motif. On peut fixer une valeur pour laquelle la variation de luminance sera considérée comme négligeable. Ainsi avec un seuil de 10 considéré comme peu visible pour l'œil humain on obtiendrait le nouveau codage indiqué dans la figure 4.2.e :

Exemple 4 :

Motifs rencontrés :

0	28	56	11	12	14
80	107	134	15	16	17
165	200	235	18	20	21

Ode du motif :

1	2	3	1	1	1
4	5	6	1	1	1
7	8	9	1	1	1

Fig 4.2.e : Exemple de motifs codés par les rangs généraux avec un seuil de 10

Cette méthode nécessite de choisir un seuil adapté aux images. Un inconvénient de cette méthode réside dans la complexité de l'algorithme, en effet la méthode des rangs

généraux demande un certain temps de calcul pour dénombrer les motifs, plus important que celui des 9 classes car la quantification se fait pour chaque motif et non au niveau de l'image. En effet si une image est un peu clair certains niveaux ne seront pas représentés dans l'image. Notre objectif ici est de concevoir un codage invariant à un étirement de l'histogramme. Cette méthode surtout si l'on ajoute le seuil se révèle donc assez gourmande en ressources.

4.2.3 Les K-Means

Le codage des 9 classes ne permet pas toujours de bénéficier des 9 niveaux considérés, le présent codage, faisant appel à l'algorithme des k-means repose sur la même idée de classification que celui des 9 classes mais contrairement à celui-ci, il est adaptatif, le codage peut être adapté à une image particulière ou à un ensemble d'images fixé. En effet plutôt que de garder des intervalles de taille fixe dans l'histogramme des niveaux de gris, on considère des intervalles plus ou moins grands pour essayer d'optimiser la ressemblance entre pixels d'une même classe et la différence entre classes comparativement à l'ensemble des niveaux présents dans l'image, on évite aussi d'avoir des classes vides. Le but de cette méthode est de regrouper en classes des données par rapport aux « centres des classes ». On prend en compte la notion de proximité entre les éléments d'une même classe.

Si on considère $\{x_1, \dots, x_n\}$ des vecteurs de \mathbb{R}^p

Classifier X , c'est définir des classes $A_1 \dots A_k$ k classes

$$X = \bigcup_{i=1}^k A_i$$

$$A_i \cap A_j = \emptyset \text{ pour } i \neq j \text{ et } i = 1 \text{ à } k \text{ et } j = 1 \text{ à } k \quad (422)$$

$$\forall i = 1 \text{ à } k, A_i \neq \emptyset \text{ et } A_i \neq X$$

A chaque classe, on associe sa fonction caractéristique χ_{A_i}

$$\text{Sup } \chi_{A_i}(xl) = 1 \text{ avec } xl \in X \text{ pour } i = 1 \text{ à } k$$

$$\text{Inf } (\chi_{A_i}(xl), \chi_{A_j}(xl)) = 0 \text{ avec } xl \in X \text{ pour } i = 1 \text{ à } k \text{ et } j = 1 \text{ à } k \quad (423)$$

$$0 < \sum_{l=1}^n \chi_{A_i}(xl) < n \text{ avec } i \text{ fixé } i = 1 \text{ à } k$$

Le nombre de classifications possibles est rapidement très grand : $\frac{1}{k!} \sum_{i=1}^k (i^k) (-1)^{k-i} - i^n$

La classification peut être représentée par une matrice :

$$\begin{matrix}
 & x_1 & x_2 & \cdots & x_n \\
 Cl_1 & \begin{pmatrix} 0 & 1 & \cdots & 1 \end{pmatrix} \\
 Cl_2 & \begin{pmatrix} 0 & 0 & \cdots & 0 \end{pmatrix} \\
 \vdots & \begin{pmatrix} \vdots & \vdots & \ddots & \vdots \end{pmatrix} \\
 Cl_k & \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}
 \end{matrix} = U \quad (424)$$

On veut minimiser la fonction objectif

$$J(U, V) = \sum_{i=1}^n \sum_{j=1}^k \chi_{ji} d(x_i, V_j)^2 \text{ avec } \chi_{ji} \text{ traduit l'appartenance de } x_i \text{ à } A_j$$

et V_j centre de la classe A_j

Pour effectuer le calcul du centre de la classe (V_j), on procède comme suit:

$$V_j = \frac{\sum_{i=1}^n \chi_{ji} x_i}{\sum_{i=1}^n \chi_{ji}} \text{ avec } V_j \text{ le centre de gravité de la classe } A_j.$$

La classification correspond au minimum de J

$$\text{Donc } J(U^*, V^*) = \min J(U, V)$$

Pour cela, on utilise un processus itératif avec un nombre de classes k fixé.

Le processus s'arrête quand il y a une convergence, c'est-à-dire que les barycentres soient à peu près fixes lors de deux itérations successives.

Cette méthode est plus efficace que celle des neuf classes mais demande un temps de calcul important. Un de ses avantages est qu'elle permet de traiter aussi bien des images en niveaux de gris que des images en couleur.

4.2.4 Contraste et gradient

Jusqu'à présent nous n'avons tenu compte que des valeurs des niveaux de gris dans le voisinage, de manière brute, comme cela est fait dans les textes où c'est la suite

d'occurrences des lettres de l'alphabet qui est utilisée. Dans le cas des images, on ne sait pas exactement comment est réalisée la perception humaine, de nombreuses théories ont été proposées [GOR04]. La valeur du pixel en un point n'est sans doute pas la seule valeur résumant l'information qui arrive à notre cerveau. Certains pensent que certains opérateurs plus sensibles aux contours orientés sont pré-établis dans notre cerveau. Nous allons donc construire un mot à partir du contexte de chaque pixel. Aussi il nous a paru intéressant de remplacer le codage des motifs bruts par un vecteur de caractéristiques qui constituera un mot. Évidemment, pour les mêmes raisons qui nous ont poussé à réduire le nombre de motifs différents possibles, nous devons limiter le nombre de caractéristiques auxquelles nous aurons recours. Nous avons choisi le contraste local et le vecteur gradient ponctuel pour caractériser une zone de l'image.

Dans cette section, nous commencerons donc par des rappels sur les notions de mesure du contraste et du gradient puis nous décrirons le principe de notre nouveau codage conduisant à de nouveaux mots sur lesquels nous verrons si la loi de Zipf s'applique.

Dans cette méthode, pour coder les motifs d'une image, associés à un pixel central du voisinage que l'on considère, on ne s'attache plus aux valeurs primitives des pixels mais on essaie d'extraire des informations d'un voisinage que nous définirons. En effet il peut être intéressant d'introduire une notion de contexte plus ou moins étendu autour d'un pixel. Les codages précédents ne considéraient les motifs que comme un ensemble de pixels ayant un simple niveau de couleur et ne tenant pas compte de la corrélation qu'il y a entre ces pixels voisins, de plus pour restreindre le nombre de motifs différents possibles nous avons dû limiter la taille de la fenêtre de voisinage choisie. Nous proposons ici, une nouvelle approche qui est de considérer un motif comme un ensemble de primitives associées à un pixel. Nous avons décidé de représenter un pixel par un motif, un mot code, et un ensemble de symboles qui intégreront le contraste dans la zone d'influence du pixel, le module du gradient et la direction du gradient calculés dans des voisinages de diverses tailles.

A côté du contraste nous introduisons aussi dans le vecteur de caractéristiques, pour identifier des voisinages n'ayant que peu de contenu comme ceux issus d'une zone très uniforme, la valeur de l'écart type calculé dans le plus petit voisinage du pixel.

Cette approche nous permet d'introduire dans notre méthode un coté plus structurel que dans les autres codages et une certaine invariance aux changements d'échelle.

- Mesure du contraste

Le contraste est défini par la variation entre le niveau de gris local et le niveau représentatif d'un voisinage plus global. En un point p , la matrice associée à l'image, étant notée I , est analysée dans une fenêtre de taille impaire. On associe au pixel central

p une mesure du contraste $C(p)$ qui fait intervenir le niveau de gris de ce point et le niveau de gris moyen $M(p)$ de son voisinage $V(p)$.

$$C(p) = \frac{I(p)}{M(p)} \quad (425)$$

Ou sous sa forme normalisée :

$$C_n(p) = \frac{I(p) - M(p)}{I(p) + M(p)} \quad (426)$$

Cette mesure est très utilisée pour le rehaussement d'image mais aussi pour la détection des contours dans [BEG89], le niveau de gris moyen des contours estimés localement dans la fenêtre d'analyse est utilisé comme indicateur pour identifier les pixels appartenant à la frontière objet/fond. Pour le codage, on peut utiliser des fenêtres correspondant à des voisinages $V(p)$ de tailles différentes, donc on aura des valeurs de contraste local différentes pour chaque point de l'image en fonction du voisinage considéré.

- Gradient

Le vecteur gradient calculé en un point de coordonnées (x,y) de l'image est obtenu en calculant le vecteur des dérivées premières de l'image par rapport aux deux coordonnées. Le résultat est une matrice des coordonnées du gradient. La valeur du module du gradient en un point reflète les variations de l'intensité du niveau de gris observée dans l'image au voisinage de ce point. Plus cette valeur est élevée, plus cette variation est forte. La valeur du module dépend de la norme utilisée. Cette amplitude peut être obtenue par le maximum des valeurs absolues des dérivées par rapport à x et à y , mais nous utilisons ici la norme euclidienne usuelle:

$$\left\{ \begin{array}{l} I_x(x, y) = \frac{\partial I_f}{\partial x}(x, y) \\ \text{et} \\ I_y(x, y) = \frac{\partial I_f}{\partial y}(x, y) \end{array} \right. \quad (427)$$

Donc, en chaque point (x, y) de l'image I , on peut calculer le module du vecteur gradient comme suit :

$$\left\{ \begin{array}{l} G = \left(I_x^2 + I_y^2 \right)^{\frac{1}{2}} \\ G = \max \left(I_x, I_y \right) \\ G = |I_x| + |I_y| \end{array} \right. \quad (428)$$

Ces calculs peuvent être faits sur l'image elle-même mais les résultats sont souvent plus significatifs quand on applique au préalable un filtre de convolution gaussien qui lisse avantageusement l'image et la transforme en une image qui serait conforme à la vision humaine.

Avec :

$$I_f = (f * I)(x, y) \quad (429)$$

Où f représente la convolution. Plusieurs opérateurs pour estimer le gradient, existent (Roberts, Sobel, Prewitt,...) qui ont globalement les mêmes propriétés. La direction du gradient est orthogonale à la frontière potentielle qui passe au point considéré. Elle est obtenue en calculant la mesure de l'angle formé par la direction horizontale et celle du gradient. On obtient la mesure de cet angle par $Arctan(I_y/I_x)$.

- Construction des mots

Après avoir rappelé les notions de contraste et de gradient nous allons les utiliser pour créer un nouveau codage. Les n-uplets de pixels sont donc remplacés par des codes de 8 valeurs, chaque mot contient 8 valeurs (niveau de gris, contraste, gradient, direction du gradient et écart type) et il représente un mot associé à un pixel de l'image à analyser.

Ng	C ₁	C ₂	G ₁	G ₂	D ₁	D ₂	σ
----	----------------	----------------	----------------	----------------	----------------	----------------	---

Fig 4.2.f : Codage d'un motif

Dans la figure 4.2.f précédente, Ng représente le niveau de gris du pixel, C₁ (resp. C₂) représente le contraste calculé sur une fenêtre 3x3 (resp. 5x5), G₁ (resp. G₂) représente la valeur du module du gradient en utilisant un voisinage de taille 3x3 (resp. 5x5), D₁ (resp. D₂) représente la direction du gradient G₁ (resp. G₂) et enfin l'écart type σ du niveau de gris dans la matrice de 3x3.

165	25	22	15	100
99	204	127	18	155
255	26	26	26	28
158	156	111	147	85
169	247	246	176	241

Valeurs Brutes	26	0,25	0,20	201	612	0,24	2,8	82
Valeurs 5-means	0	1	0	3	4	1	2	5

Fig 4.2.g : Exemple d'extraction d'un mot à partir d'un pixel et de ses voisinages 8 connexes de taille 3x3 et 5x5

En considérant toutes les valeurs possibles avec la précision informatique des réels pour Ng, C_i, G_i, D_i et σ, le nombre de motifs possibles devient trop grand et la probabilité d'apparition d'un même motif plusieurs fois dans l'image serait très faible. En effet le nombre de mots composant une image est égal au nombre de pixels de l'image par exemple pour une image de 800 pixels sur 600, on ne dispose que de 480 000 motifs. Le nombre de motifs différents dans ce type de codage est égal au nombre d'éléments

possibles dans la quantification de chacun des paramètres, à la puissance 8.

Pour réduire le nombre de motifs possibles, nous appliquons à chacune des grandeurs composant le motif une classification k-means en 5 classes (5 valeurs possibles pour chaque grandeur), ce qui donne un nombre de motifs maximal de $5^8=390625$ motifs différents.

INTERPRÉTATION DU GRAPHE DE ZIPF

5 Interprétation du graphe de Zipf

Dans cette section nous nous attacherons à appliquer nos différentes méthodes sur des images les plus variées possible pour s'assurer de la généralité de nos résultats. Les images seront des images naturelles, des images de synthèse ainsi que des numérisations de documents anciens. Les courbes de Zipf et de Zipf inverse seront analysées et on proposera d'extraire des descripteurs caractérisant les images à l'aide des paramètres issus de ces lois de Zipf et de Zipf inverse modélisant les distributions contenues dans l'image. Nous distinguerons les conclusions en fonction du mode de codage des mots utilisé.

5.1 Les Rangs généraux

Nous allons appliquer la méthode de codage des rangs généraux, celle décrite par BI [BI97] et dans la section 4.2.2 de ce document, se nommant méthode par vecteur de rangs généraux où deux pixels de niveaux de gris égaux appartenant à un même motif auront le même rang dans ce motif.

Ce qui nous donne que si l'on considère $f_i, i=0, \dots, n$ les niveaux de gris des $n+1$ pixels du motif et $r_i, i=0, \dots, n$ les rangs des $n+1$ pixels du motif, on ordonne les f_i dans l'ordre croissant. Le rang est donné par :

$r_{i_0} = 0$ pour le plus petit niveau de gris

on affecte ensuite les rangs avec la condition :

si $f_{i_n} = f_{i_n'}$ alors $r_{i_n} = r_{i_n'}$

Le choix de la forme de notre motif considéré sera une matrice 3x3 correspondant au voisinage 8-connexe. Dans une première partie nous observerons les courbes de la loi de Zipf classique, puis dans une seconde partie nous nous attacherons à considérer les courbes de Zipf inverse.

5.1.1 Les rangs généraux et la loi de Zipf

L'application de notre méthode sur des images en niveaux de gris nous donne pour la courbe de Zipf un ensemble de points qui peuvent raisonnablement être approximés par une droite linéaire, la loi de Zipf est donc vérifiée selon ce codage.

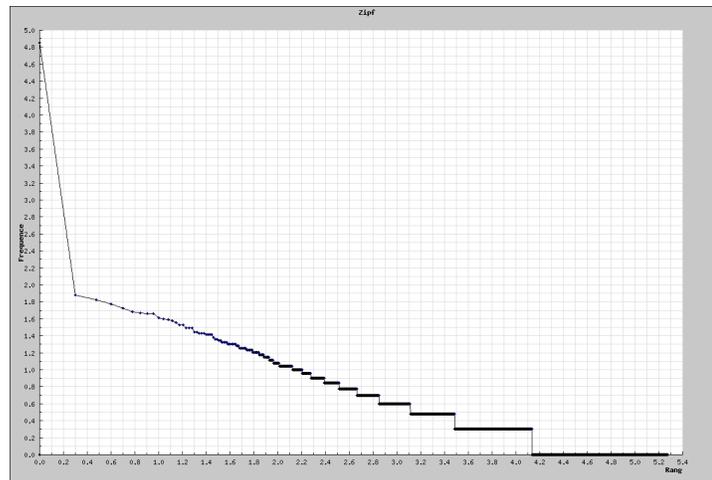


Fig 5.1.a : extraction de la courbe de Zipf d'une image par la méthode des rangs généraux

Le coefficient de corrélation linéaire des données ré-échantillonnée est très proche de 1. On remarque que sur cette image fortement structurée d'un bâtiment en construction typique de la réalisation industrielle moderne qui donne une image pleine de ligne parfaitement verticales, la courbe est assez linéaire sauf pour le premier motif correspondant à un motif totalement uniforme. Essayons d'interpréter ce résultat. Dans cette image de taille 615 pixels sur 460 comportant donc 282900 pixels, la répartition des motifs est celle –ci pour les douze premiers.

Motif	Rang	Fréquence
0 0 0	1	70919
0 0 0		
0 0 0		
3 0 0	2	76
2 0 0		
1 0 0		
1 0 0	3	67
2 0 0		
3 0 0		
6 3 0	4	60
7 4 1		
8 5 2		
1 1 1	5	53
1 1 1		
1 1 0		
2 0 0	6	48
3 0 0		
1 0 0		

Motif	Rang	Fréquence
1 1 0	7	47
1 1 1		
1 1 1		
0 1 1	8	46
1 1 1		
1 1 1		
2 0 0	9	46
1 0 0		
3 0 0		
1 0 0	10	41
3 0 0		
2 0 0		
2 5 8	11	40
1 4 7		
0 3 6		
3 0 0	12	39
1 0 0		
2 0 0		

Tab 5.1.1 : liste des douze premiers motifs de la figure 5.1.a

Le motif le plus fréquent est le motif uniforme. Ceci s'explique car l'image est fortement résolue et l'impression visuelle cache cette uniformité à moins que ce ne soit la saturation du ciel en haut à gauche. On remarque ensuite que les motifs les plus présents dans l'image correspondent pour la plupart à des motifs de structures particulières, d'ailleurs on remarque que celui qui revient en 2, 3, 6, 9, 10 et 12^{ème} rang correspond à un motif de contour vertical, à gauche d'une surface homogène. Si on s'intéresse à l'image dont sont extraits les motifs, on peut voir que celle-ci contient de très nombreuses structures verticales. On notera par exemple les arêtes des immeubles ainsi que les échafaudages verticaux. Es mots étant associés à chaque pixel, on peut remarquer que les motifs considérés sont recouvrants ce qui peut expliquer en partie certaines symétries.



Fig 5.1.b : Image naturelle avec beaucoup de contours

La photo ci-dessus comporte aussi de nombreuses structures rectilignes apparaissant sur la mosaïque carrelée du Bouddha, mais les contours ne sont plus strictement verticaux ou horizontaux comme sur l'image de la 5.1.a. Cette image est intéressante car elle représente toujours une réalisation humaine mais cette fois faite sans recourir à une machine, uniquement réalisé par la main de l'homme, de plus contrairement à l'image 5.1.a le fond de celle-ci est un paysage naturel et non un ciel très uniforme. Cette photo a été prise par le même capteur que la première avec la même résolution. Si on en extrait la courbe de Zipf on s'attend à trouver le même phénomène que celui constaté dans la précédente photo.

La courbe de Zipf est présentée en Figure 5.1.c.

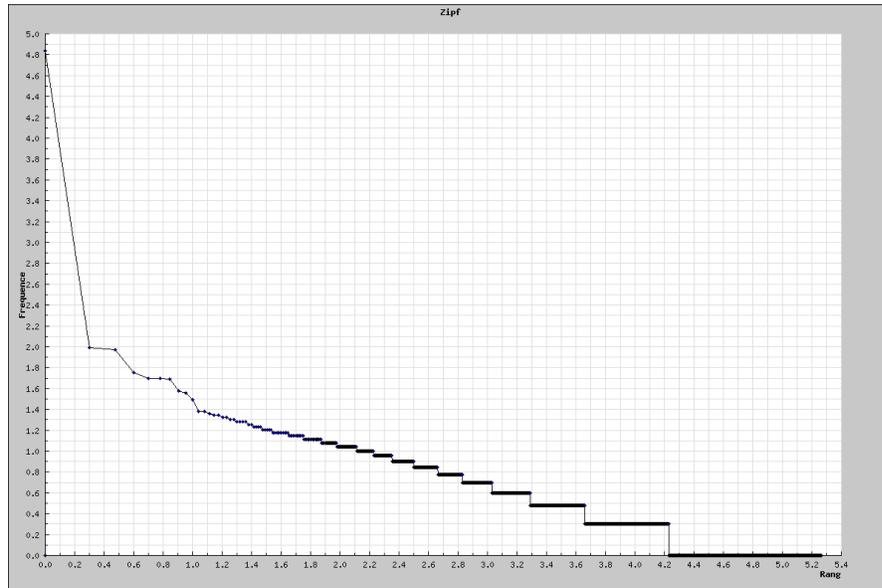


Fig 5.1.c : Courbe de Zipf avec le codage des rangs généraux de la figure 5.1.b

La courbe associée à cette image est proche de celle de la figure 5.1.a, on remarque ainsi le même motif le plus fréquent apparaissant majoritairement au sein de l'image, puis une courbe très linéaire par la suite avec, en fin de courbe, un effet d'escalier. Le fond de l'image représentant un paysage naturel n'a eu que peu d'impact sur le premier motif et l'on observe le même décrochage que sur l'exemple de l'image 5.1.a. Ainsi on peut conclure que l'hypothèse de la saturation ou de la grande uniformité du fond retenue précédemment ne se vérifie pas ici. Pour nous éclairer un peu plus sur le phénomène observé intéressons-nous à la distribution des motifs les plus fréquents et en particulier au motif le plus fréquent qui semble caractériser les courbes de Zipf extraites avec le codage des rangs généraux.

Cette distribution des motifs les plus fréquents se trouve dans le tableau suivant :

Motif			Rang	Fréquence
0	0	0	1	69325
0	0	0		
0	0	0		
1	0	0	2	98
2	0	0		
3	0	0		
3	0	0	3	94
2	0	0		
1	0	0		
2	0	0	4	57
1	0	0		
3	0	0		
1	0	0	5	50
3	0	0		
2	0	0		
3	0	0	6	50
1	0	0		
2	0	0		

Motif			Rang	Fréquence
2	0	0	7	49
3	0	0		
1	0	0		
0	1	2	8	38
3	4	5		
6	7	8		
8	7	6	9	36
5	4	3		
2	1	0		
6	3	0	10	31
7	4	1		
8	5	2		
2	5	8	11	24
0	4	7		
1	3	6		
1	0	2	12	24
3	4	5		
6	7	8		

Tab 5.1.b : liste des douze premiers motifs de la figure 5.1.a

Le modèle introduit par les rangs généraux donne donc des résultats similaires à ceux observés par la loi de Zipf dans des phénomènes monodimensionnels. En fait on a perdu l'organisation globale des pixels de l'image pour ne plus considérer que le sac de mots comme cela est fait avec les signaux monodimensionnels. Sur le codage des rangs généraux on peut dire que la loi de Zipf s'applique pleinement sur les images non

naturelles représentant des structures organisées par la main de l'homme avec l'aide ou non de machine. Seul petit bémol à cette observation est la fréquence très élevée remarquée sur le motif qui caractérise la courbe extraite par la méthode des rangs généraux. Il semble que même si une image ne contient pas à priori de grandes zones uniformes visibles au premier regard, le codage par les rangs généraux soit tout de même capable d'extraire un grand nombre de motifs uniformes. Il est vrai que si l'on regarde de plus près l'image et si l'on rapporte cela à la taille finalement assez réduite de notre fenêtre de motif, il n'est pas étonnant de constater ce phénomène.

Nous allons maintenant appliquer le codage des rangs généraux sur une image de lettrine issue d'un document ancien et observer si celle-ci se comporte de la même façon qu'avec une image naturelle vis-à-vis de la loi de Zipf.



Fig 5.1.d : Image de lettrine issue d'un document de la Renaissance du CESR

Si l'on extrait la courbe de Zipf de l'image présentée en figure 5.1.d, à l'aide du codage des rangs généraux, avec un masque de motif de 3x3. On obtient la courbe illustrée en figure 5.1.e :

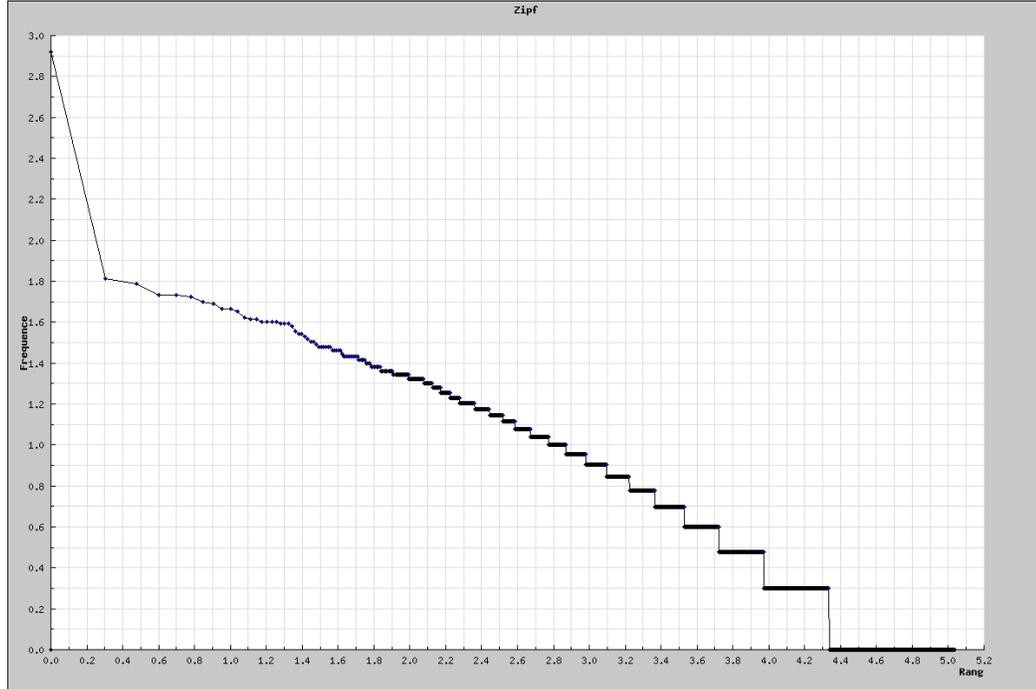


Fig 5.1.e : Courbe de Zipf extraite de la figure 5.1.d à l'aide du codage des rangs généraux

Pour les images de documents anciens on remarque que la courbe a la même allure que celles des images non documentaires.

Quand on extrait la courbe de Zipf de plusieurs images à l'aide du codage des rangs généraux on obtient des courbes plutôt similaires. Ainsi le motif apparaissant le plus fréquemment dans l'image est le motif uniforme. Sa fréquence est vraiment plus importante que celle des autres motifs. Puis on observe que ce sont les motifs représentant des contours rectilignes qui apparaissent, suivis ensuite par les autres motifs représentant plutôt des surfaces ou des textures. En effet le codage a été construit pour mettre en évidence les ruptures, donc les contours.

On peut affirmer que, si l'on fait abstraction du motif uniforme, la forme de la courbe permet d'affirmer que la répartition des pixels dans l'image suit la loi de Zipf et que l'on pourra donc caractériser celle-ci à l'aide de l'exposant de la loi, c'est-à-dire le coefficient directeur de la droite de régression de la courbe associée.

Le codage par rangs généraux, par construction, s'attache aux contours et donc à la structure d'une image. Ainsi, on met en évidence la présence d'objets et c'est la direction des différents contours, leurs proportions que l'on résume dans la courbe de Zipf. L'exposant traduit ces relations. Ainsi, ce codage donne à notre méthode une approche à

caractère moins purement statistique.

On notera de plus, comme il avait été remarqué dans la section concernant la loi de Zipf et ses diverses applications, en regardant plus précisément les courbes, l'apparition de trois parties différentes dans la courbe et ceci de façon assez remarquable. Une première partie étant composée du motif uniforme, une deuxième très rectiligne puis une troisième en escalier.

Ces similarités dans la forme générale de la courbe de Zipf n'est pas un très bon point pour la recherche par le contenu. Et on voudrait s'attacher à trouver des méthodes plus discriminantes et variant d'un type d'image à l'autre. En effet si l'on cherche des choses similaires il faut aussi pouvoir écarter de façon assez tranchée les images non similaires entre elles. Or ici on peut constater qu'une image naturelle peut avoir une courbe assez proche d'une image de document ancien.

Un autre point négatif est que le nombre de primitives que l'on peut extraire de la courbe est relativement faible. En effet on peut, pour caractériser une image, extraire la pente du segment linéaire ce qui donne une seule primitive et qui risque d'être proche de celle d'autres images. On pourrait aussi tester la fréquence du motif le plus uniforme en considérant des fréquences normalisées.

De plus l'effet d'escalier en fin de courbe est très important et on le doit au fait que beaucoup de motifs n'apparaissent qu'une fois au sein de l'image, on peut même considérer qu'un motif ne sera réellement pris en compte que s'il apparaît au moins deux fois dans l'image.

5.1.2 Les rangs généraux et la loi de Zipf inverse

Avec la méthode de codage des rangs généraux on peut non seulement tracer la courbe de Zipf classique mais aussi celle de Zipf inverse qui considère, comme expliqué dans la section 3.2.5, uniquement les motifs de faible fréquence.

Ainsi nous devrions pouvoir remarquer comme pour la loi de Zipf classique une certaine linéarité dans la courbe de Zipf inverse extraite d'une image.

Si nous reprenons l'image de la figure 5.1.a nous obtenons la courbe présentée en figure 5.1.f

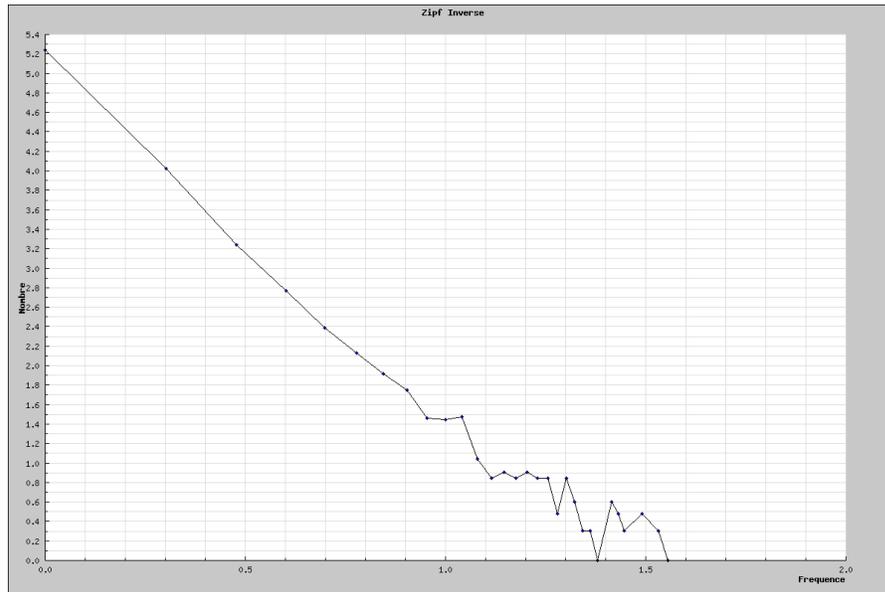


Fig 5.1.f : Courbe de Zipf inverse extraite avec le codage des motifs en rangs généraux

La courbe est composée de deux parties la première qui respecte une remarquable linéarité puis une deuxième un peu plus folle en « zigzag ». Ce phénomène avait déjà été noté dans les études menées par G.K. Zipf qui affirme que la loi de Zipf inverse n'est vérifiée que pour les motifs les moins fréquents et non pour tous les motifs extraits d'une image. On remarque pour notre part que la loi de Zipf inverse est ici pleinement vérifiée et si l'on extrait le coefficient directeur de la droite de régression que forment les dix premiers motifs les moins fréquents on obtient -3,8985 avec un coefficient linéaire proche de 1.

Si nous extrayons la courbe de Zipf inverse de l'image présentée en figure 5.1.d nous obtenons le graphique présenté figure 5.1.g

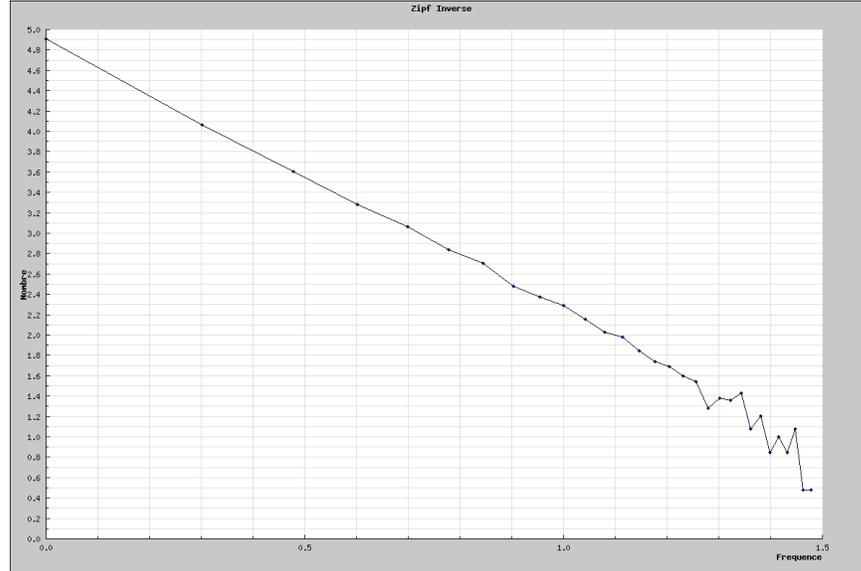


Fig 5.1.g : Courbe de Zipf inverse de la lettre présentée en figure 5.1.d

Ce que l'on peut noter c'est que la linéarité est plus longue dans cette image issue d'un document que dans une photographie, mais il est à noter que le coefficient directeur de la droite de régression est de -3,625 soit un nombre assez proche de ce que nous avons trouvé auparavant.

Nous avons extrait les courbes de Zipf inverse d'un grand nombre d'images et nous avons toujours pu faire le constat de la présence de la même linéarité dans toutes les courbes tracées. Il semble que la loi de Zipf inverse soit pleinement vérifiée pour les images lorsqu'elle est extraite à l'aide du codage des rangs généraux. .

5.2 Les classes

Dans cette section nous allons observer ce que donne l'essai d'application de la méthode de quantification des couleurs au sens large du terme par un k-mean pour réduire le nombre de motifs, appliqué à divers types d'images. Pour notre application, nous gardons une matrice de 3x3 pour extraire les motifs ; ce ne sera que le nombre de classes de niveaux de gris dans un premier temps, puis de l'espace RGB qui variera, nous lui donnerons une valeur comprise dans une fourchette allant de 2 à 9 pour garder un nombre de motifs raisonnable. Les images observées seront choisies en fonction de leur nature et des caractéristiques de celles-ci. Nous essayerons de montrer comment la courbe de Zipf se construit et si la loi est bien vérifiée. Les images présentées dans la première partie de cette section sont originalement codées en RGB mais la classification se fera sur les niveaux de gris de l'image, c'est-à-dire qu'avant tout traitement on fera une moyenne pondérée des valeurs RGB afin de récupérer un histogramme de 256 valeurs.

Ce n'est donc pas un traitement d'image couleur. Ce n'est que dans un deuxième temps que nous utiliserons une classification directement faite sur les valeurs RGB de l'image sans qu'une quelconque conversion ne soit effectuée. Dans une dernière partie nous nous intéresserons à la courbe de Zipf inverse extraite grâce à ces codages.

5.2.1 Les classes et la loi de Zipf

Si nous reprenons les images utilisées dans la section précédente par exemple celle représentant des immeubles :



Fig 5.2.a : Même image que dans la figure 5.1.a en niveaux de gris

Nous obtenons les courbes de Zipf suivantes en fonction du nombre de classes :

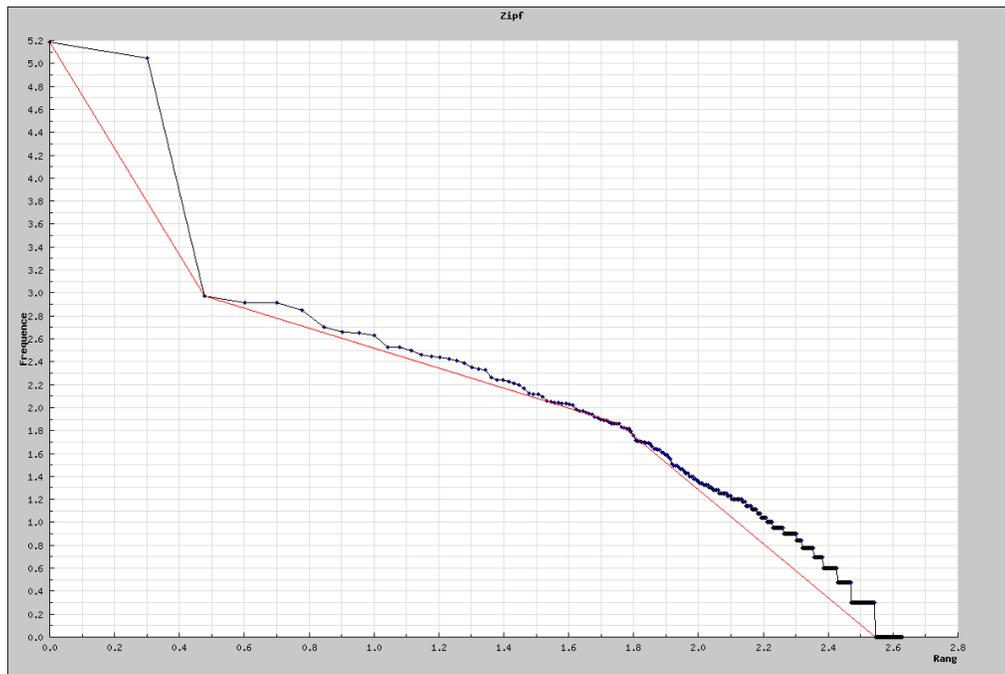


Fig 5.2.b : Courbe de Zipf avec K-means K=2

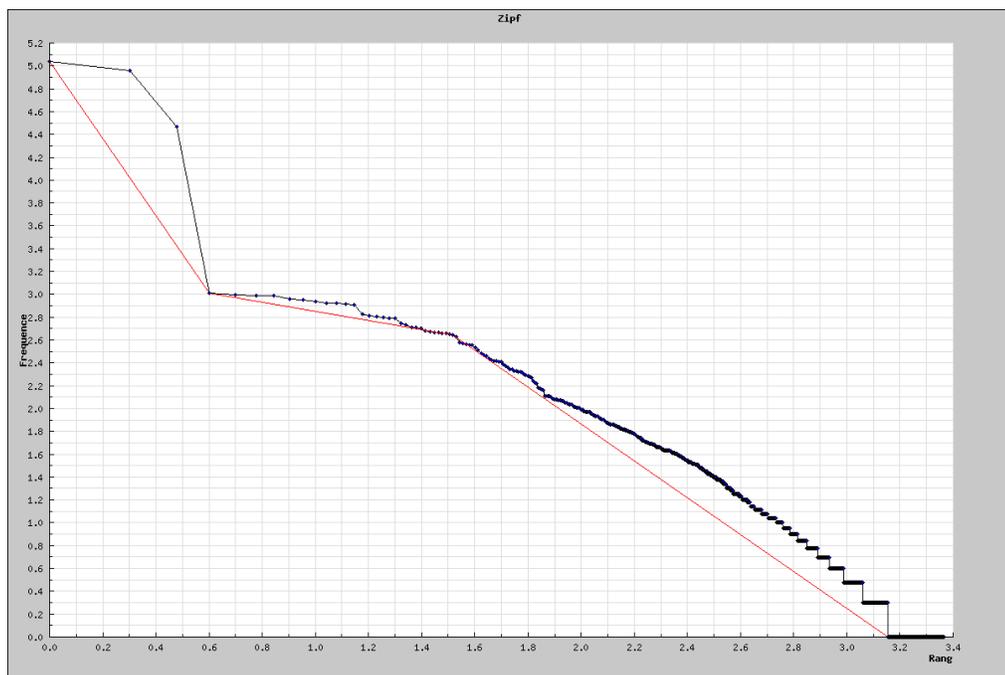


Fig 5.2.c : Courbe de Zipf avec K-means K=3

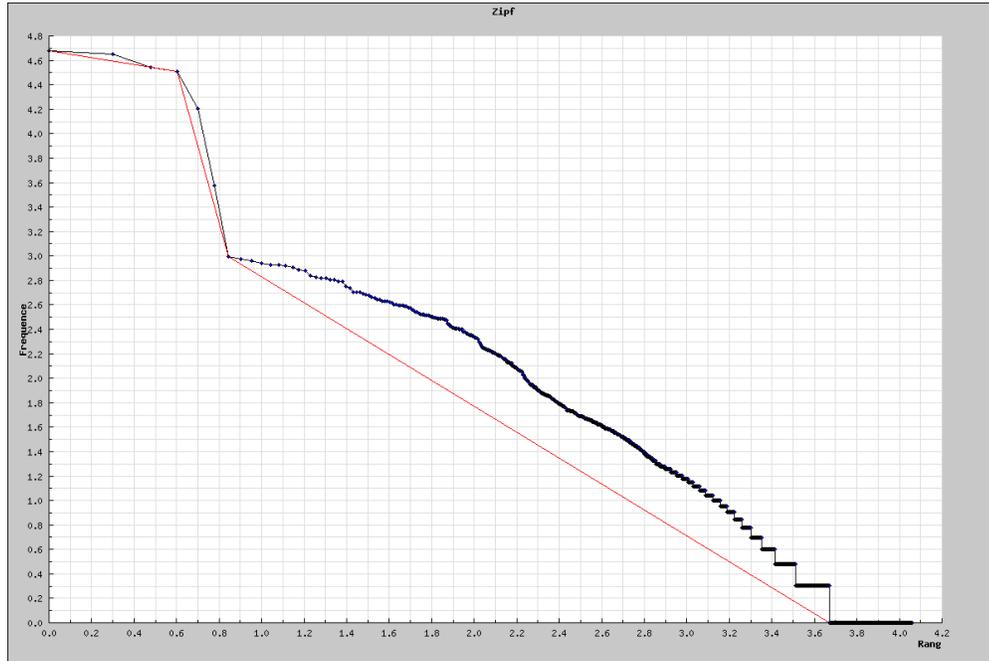


Fig 5.2.d : Courbe de Zipf avec K-means K=6

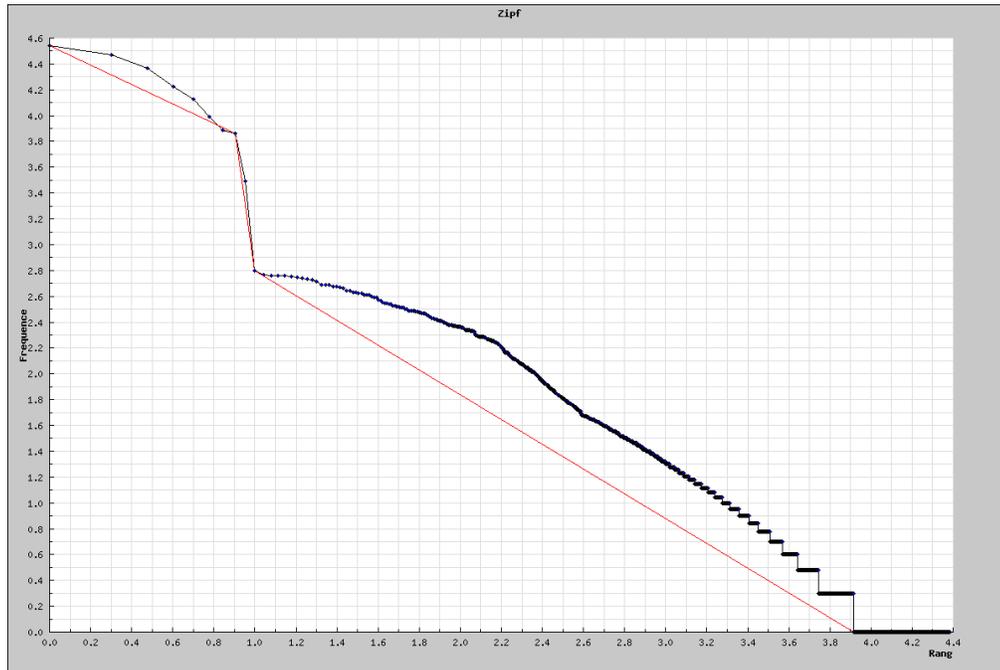


Fig 5.2.e : Courbe de Zipf avec K-means K=9

Si nous observons les courbes extraites nous voyons que nous pouvons conclure que la loi de Zipf ne s'applique pas. En effet nous ne retrouvons pas un segment linéaire mais trois.

Niveau de K-means	Coefficient Directeur 1	Coefficient Directeur 2	Coefficient Directeur 3
2	-4.630	-0.875	-2.352
3	-3.369	-0.394	-1.616
6	-0.284	-6.218	-1.060
9	-0.754	-10.971	-0.959

Tab 5.2.1 : Comparaison des différents coefficients directeurs des trois droites de régression en fonction du nombre de classes.

On remarque dans cet exemple que les résultats pourtant issus d'une même image sont très différents d'un niveau de classification à un autre. En revanche ce que l'on peut constater c'est que cette découpe en trois segments linéaire est un trait caractéristique de la méthode de codage des motifs par les k-means.

Partant de ce constat il nous est maintenant possible de vérifier si cette caractéristique se retrouve dans les autres types d'image ou s'il n'est présent que dans les images naturelles.

Vérifions l'observation pour une image de document ancien.



Fig 5.2.f : Lettrine issue d'un ouvrage datant de la renaissance

Nous allons appliquer sur cette image notre méthode en faisant varier le nombre de classes considérées et observer les courbes résultantes.

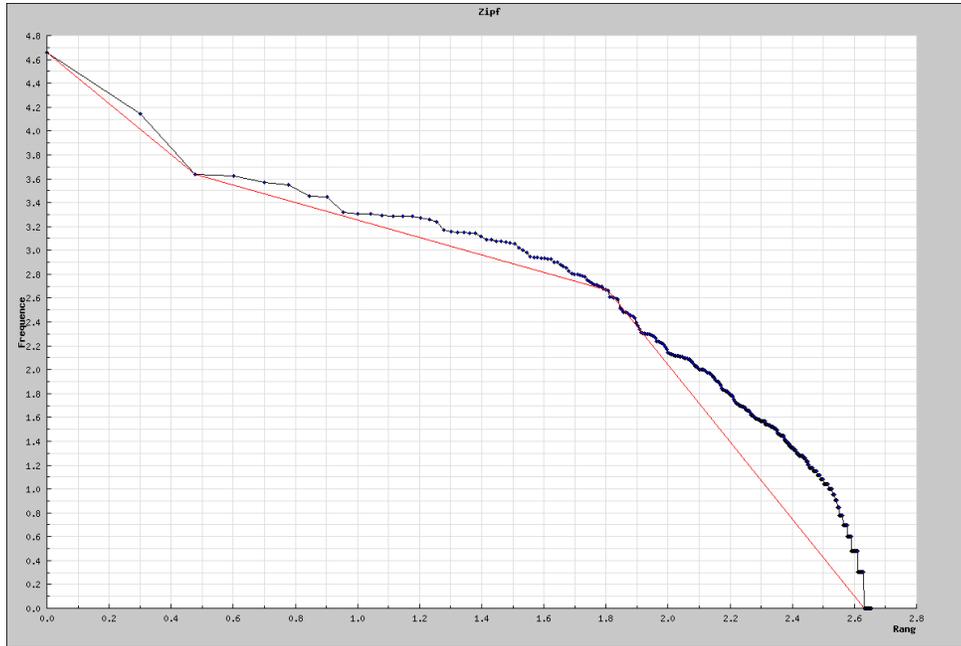


Fig 5.2.g : Courbe de Zipf avec K-means K=2

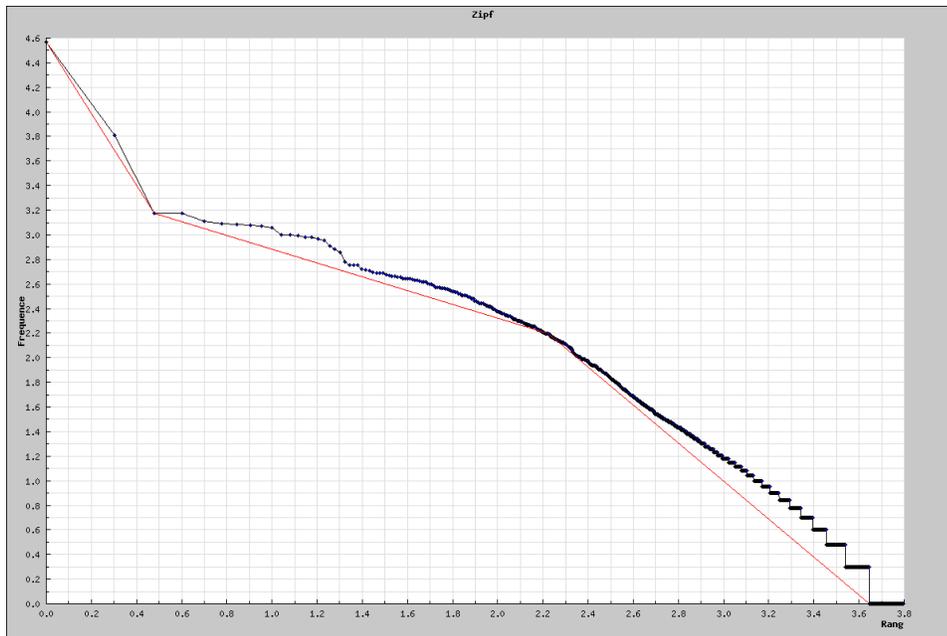


Fig 5.2.h : Courbe de Zipf avec K-means K=3

Sur les images de document les courbes de Zipf extraites à l'aide des K-means avec K

supérieur à 3, conduisent à des courbes de Zipf moins significatives. Les courbes apparaissent comme n'ayant pas de structure caractéristique et on observe des décrochages entre certains motifs qui font perdre la linéarité à la courbe. Ce phénomène n'est pas étonnant car la nature même des images de documents anciens ne justifie pas de classer les couleurs en un nombre supérieur à 3. On peut dire que les images de documents anciens que nous utilisons ne sont pas comme d'habitude binaires mais plutôt ternaires. Il y a entre le noir et le blanc une couleur intermédiaire grise que l'on retrouve autour des contours et dans certaines textures. Ce troisième niveau entre le blanc et le noir fait parti intégrante de l'image et apporte une information essentielle sur la structure de celle-ci. Ce troisième niveau de gris pourrait être apparenté à l'effet obtenu lorsqu'on applique sur une image un filtre d'anti-aliasing qui permet de fondre la dureté des contours.



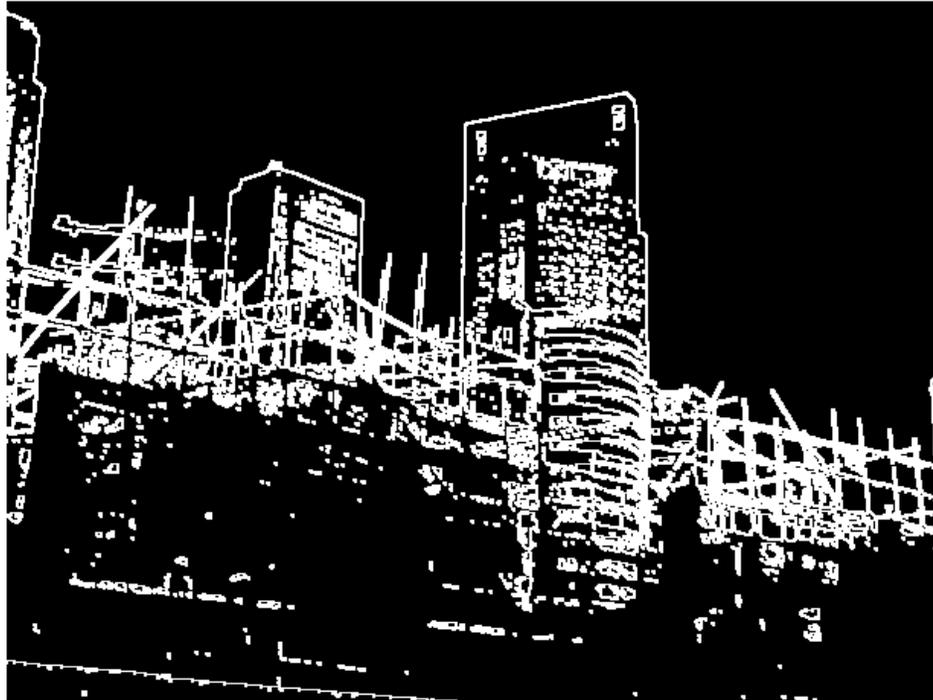
Fig 5.2.i : Partie d'une image de lettrine agrandie

De ce fait l'algorithme des k-means avec $k=3$ apparait naturellement le mieux adapté à ce type d'image.

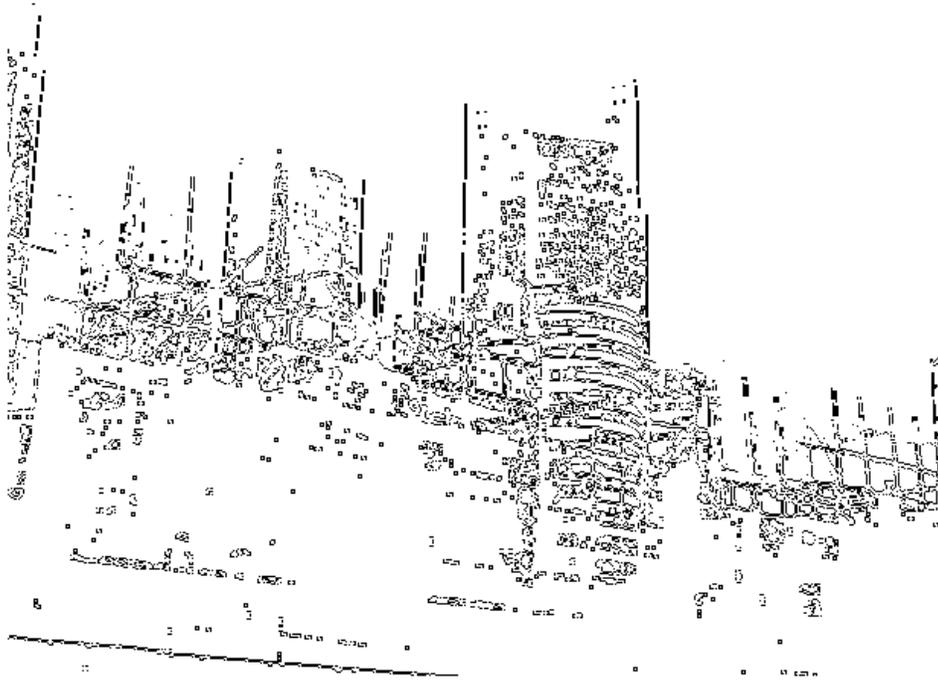
L'observation de la courbe de Zipf extraite à l'aide de la méthode des K-means avec $k=3$ fait apparaître de nouveau une courbe en trois parties. Pour mieux appréhender ce que représente les trois différents segments de la courbe de Zipf on peut regarder quels

pixels sont concernés par chacune des trois parties de la courbe.

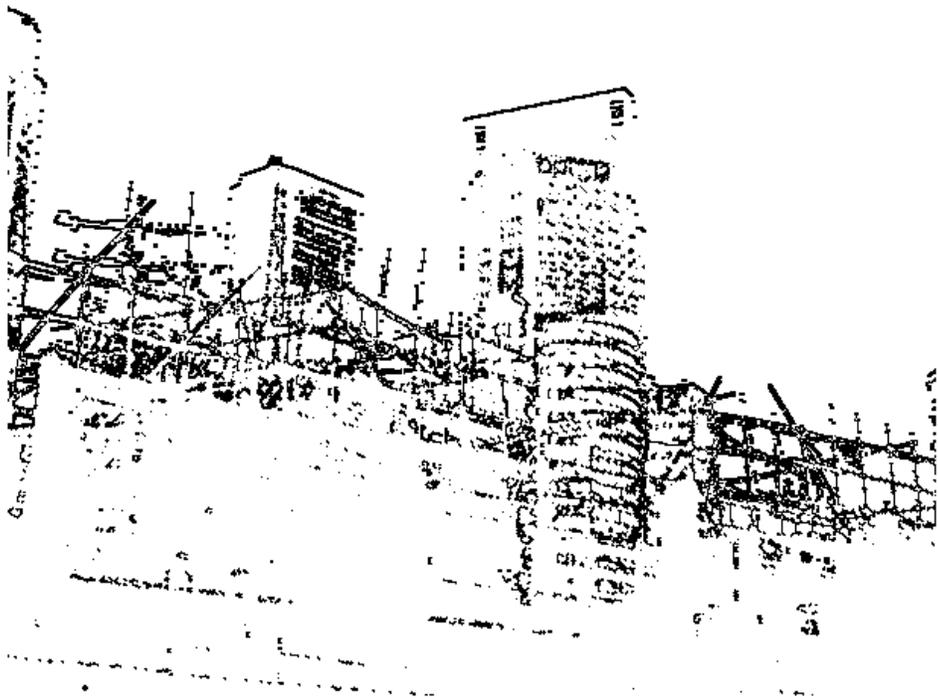
Si nous reprenons l'image de building en figure 5.2.a et que nous extrayons les pixels de chacun des segments nous obtenons les trois images de la figure 5.2.j



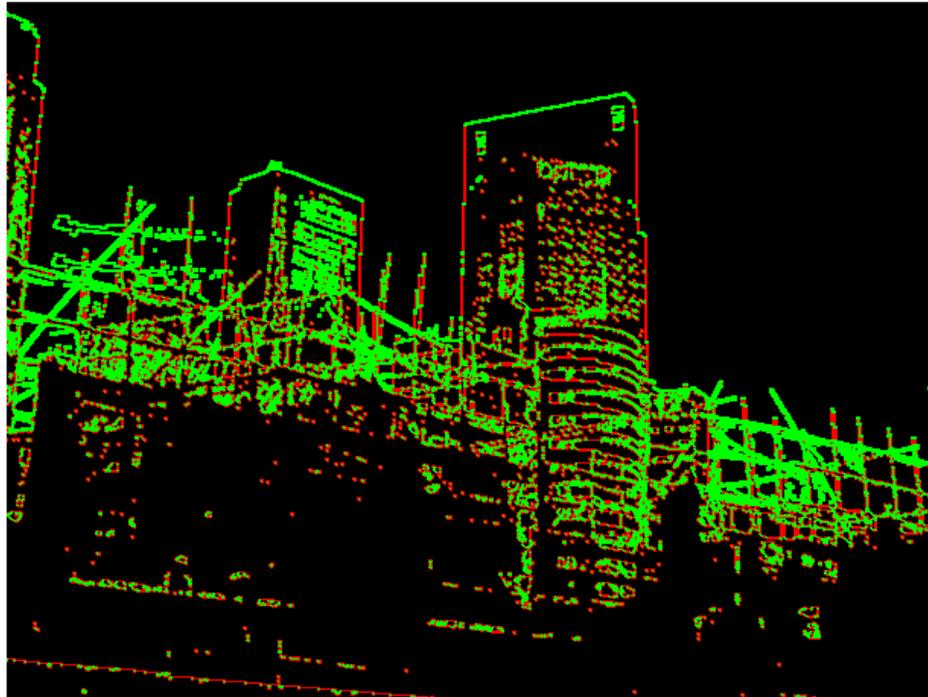
Pixels concernés par le segment 1 $k=3$ (les pixels sont en noir)



Pixels concernés par le segment 2 $k=3$



Pixels concernés par le segment 3 $k=3$



Les 3 couches sur une même image noir =segment1, rouge=segment2, vert=segment3

Fig 5.2.j : décomposition de l'image en trois couches de pixels

L'analyse nous montre que les différents segments concernent des pixels très caractéristiques de l'image. On remarque que les pixels centraux issus des motifs composant le premier segment de la courbe sont des pixels qui composent les surfaces homogènes. Ainsi, les motifs composant le ciel de l' image ainsi que les murs des buildings et l'intérieur de l'échafaudage se trouvent être les plus fréquents de l'image. Ensuite les motifs qui composent le deuxième segment sont ceux qui composent les contours dans l'image.

La composition du segment numéro 3, en vert est un peu plus difficile mais il semble que les motifs le formant soient des motifs de "surface de contours" entendez des motifs se trouvant à l'intérieur d'un ensemble de motifs issus du segment numéro 2. La figure 5.2.k qui est un agrandissement de la dernière image de la figure 5.2.j permet de mieux comprendre cette notion.

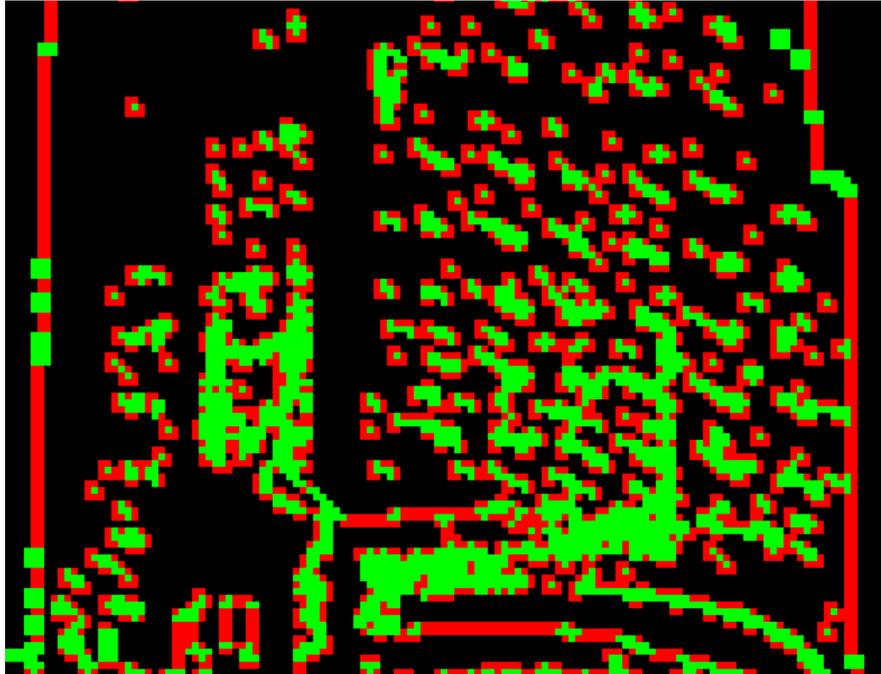
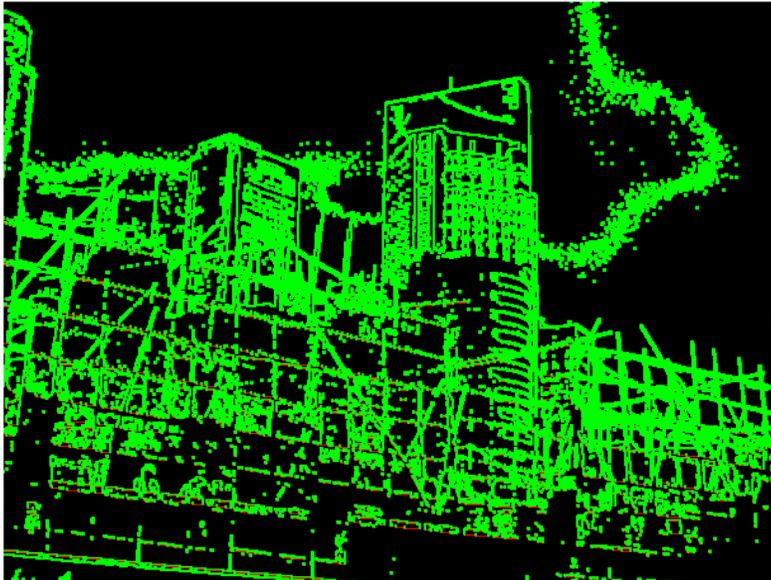
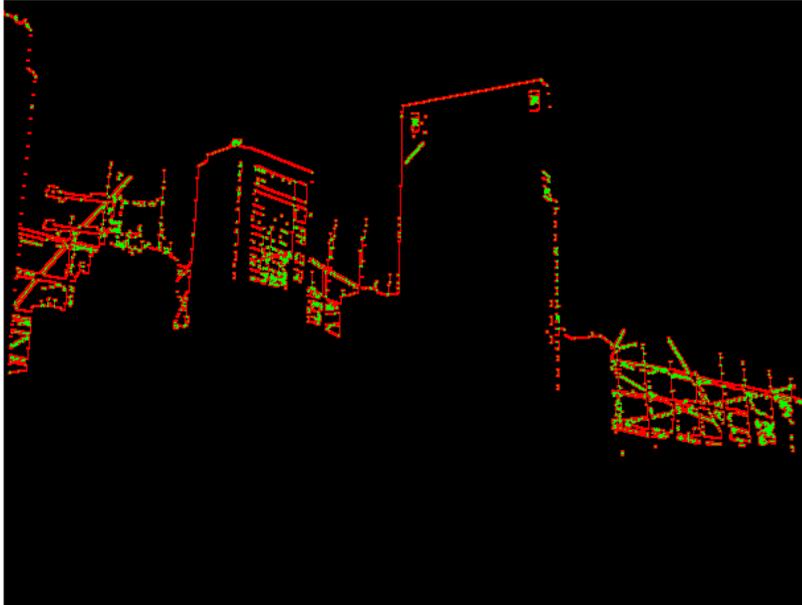


Fig 5.2.k : Agrandissement de l'extraction des couches.

Si l'on observe les couches extraites avec des niveaux de K-means différents de 3 on se rend compte que les résultats sont moins bons et leurs qualités dans l'interprétation décroissent avec l'augmentation de l'entier naturel k . Ceci est illustré en figure 5.2.l.



K=5



K=2

Fig 5.2.1 : décomposition de l'image en trois couches de pixels avec variation de K

Ce que l'on constate quand on fait varier K c'est qu'il existe un nombre de classes qui optimise la qualité de l'interprétation des résultats, ainsi, pour notre exemple on peut noter que la valeur pivot de K semble être 3. En effet, si l'on utilise une valeur inférieure à 3 le résultat obtenu n'est pas véritablement satisfaisant, beaucoup de pixels correspondant à des contours ont été reconnus comme des motifs de zone ayant une texture uniforme. On obtient une image résultat très uniforme avec quelques contours correspondant aux contours les plus prononcés de l'image originale, c'est-à-dire les pixels dont le contraste avec leurs voisins est le plus grand.

En revanche si l'on donne une valeur supérieure à 3 au paramètre K, traduisant le nombre d'étiquettes possibles à attribuer à un pixel, nous voyons apparaître des contours peu visibles à l'œil humain et surtout peu pertinents. Cette observation n'est pas nouvelle et avait déjà été notée, en effet théoriquement pour que la loi de Zipf s'applique pleinement il faut pouvoir extraire du phénomène observé un nombre de motifs conséquent. Que serait par exemple la courbe de Zipf d'une image ne comportant que

des pixels d'un même niveau de gris on serait dans l'impossibilité de tracer la courbe réduite à un seul point. L'exemple contraire mènerait aussi à un résultat peu probant si, par exemple une image comportait autant de pixels que de motifs extraits chacun ayant la même fréquence. Que serait et surtout que signifierait la courbe de Zipf d'un tel phénomène ? On peut donc conclure qu'il existe un point d'équilibre au nombre de motifs possibles pouvant être extraits.

Il semblerait que pour les images en niveau de gris ce nombre de motifs soit celui trouvé quand on applique un 3-means sur l'image ce qui donne 19683.

Pour les images en niveaux de gris, il se révèle que ce nombre soit celui que l'on trouve quand on classifie les images en trois niveaux de gris avec une matrice de motif de 3 pixels sur 3.

Pour étayer notre observation nous pouvons observer la répartition des premiers motifs les plus fréquents ainsi que leur valeur quand on utilise un codage utilisant un k-means avec un paramètre k ayant une valeur de 3 sur l'image présentée en figure 5.2.a.

Motif			Rang	Fréquence	Motif			Rang	Fréquence
2	2	2	1	108788	1	1	1	7	970
2	2	2			1	1	1		
2	2	2			0	0	0		
0	0	0	2	91177	1	1	1	8	911
0	0	0			0	0	0		
0	0	0			0	0	0		
1	1	1	3	29337	1	1	0	9	887
1	1	1			1	1	1		
1	1	1			1	1	1		
0	0	1	4	1018	0	0	0	10	866
0	0	0			0	0	0		
0	0	0			1	0	0		
0	0	0	5	991	0	0	0	11	838
1	1	1			0	0	0		
1	1	1			1	1	1		
1	1	1	6	978	1	2	2	12	24
1	1	1			1	2	2		
0	1	1			1	2	2		

Fig 5.2.m : Valeur des 12 premiers motifs de l'image codée avec un kmean avec k=3 et une matrice de 3x3

L'observation de la figure 5.2.m nous permet de vérifier ce que l'on remarquait plus tôt. Les trois motifs formant le premier segment sont des motifs uniformes issus des surfaces homogènes de l'image. Puis les motifs suivants sont issus de pixels de contours et de coins. Si l'on observe les motifs de chaque segment on obtient la figure 5.2.n



Fig 5.2.n : Agrandissement de l'extraction des couches d'une image de lettrine.

Si l'on observe attentivement la figure 5.2.n on s'aperçoit que les motifs faisant partie du deuxième segment sont en fait extraits de pixels de contours que l'on peut qualifier de contours flous. Ces pixels sont observables sur la figure 5.2.i où l'on remarque que les pixels avec un niveau de gris moyen c'est-à-dire ni trop proche du noir ni trop proche du blanc sont accolés aux pixels de contours noirs. Ce sont eux qui composent le deuxième segment de la courbe. C'est ce qui ressort aussi de la figure 5.2.m qui montre des motifs ayant une majorité de cellules de valeurs 1 qui correspond à une valeur moyenne de niveau de gris.

5.2.2 Les classes couleur

La méthode de codage des motifs par les K-means peut très facilement s'adapter à des images, non plus en niveaux de gris mais en couleur. Cette adaptation ne se fait qu'au prix d'un temps de calcul un peu plus long.

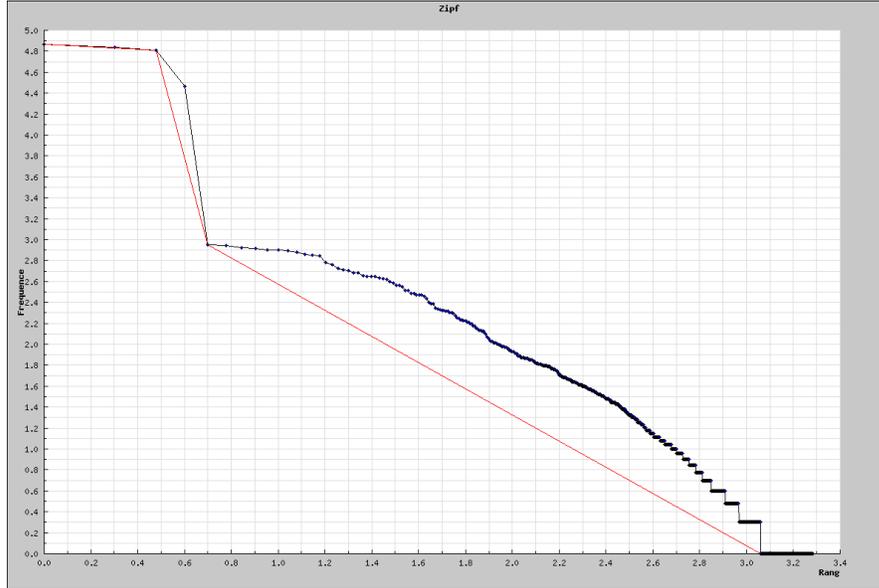


Fig 5.2.o : courbe de zipf extraite de l'image couleur de la figure 5.1.a

On remarque le même phénomène qu'avec des images en niveaux de gris, c'est-à-dire l'apparition de trois parties caractéristiques. On peut comme auparavant observer la répartition en trois couches des pixels issus des motifs de chacun des segments.

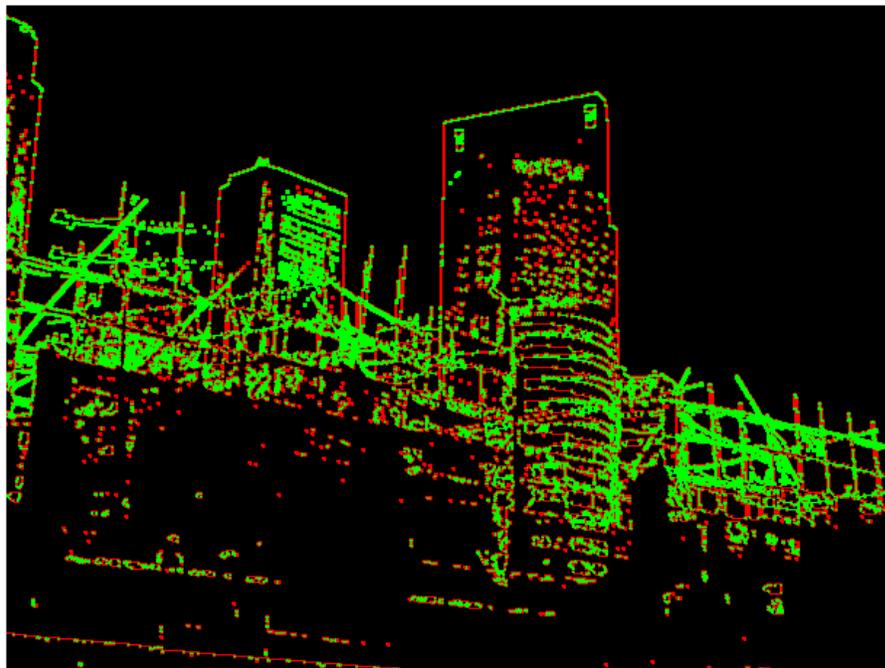
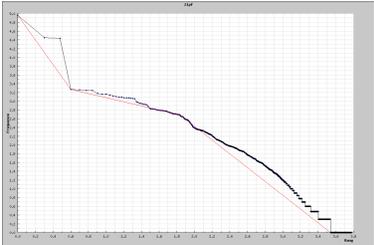
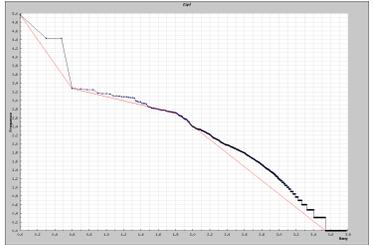
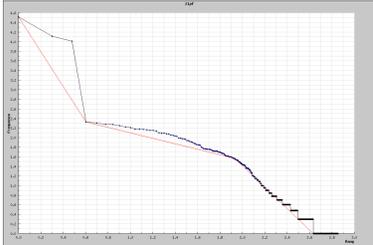
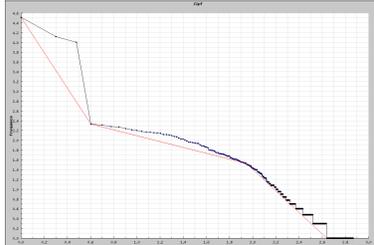


Fig 5.2.o : décomposition de l'image en trois couches de pixels avec un 3-means couleur

L'image présentée en figure 5.2.o est composée de pixels de trois couleurs, le noir correspondant aux pixels formant la première partie de la courbe de Zipf associée, le rouge aux pixels de la deuxième et le vert aux pixels de la troisième.

On remarque que l'image extraite est très proche de celle extraite à partir de la même photographie mais en niveaux de gris et présentée en figure 5.2.j. Ce qui tendrait à faire penser que la couleur n'a que peu d'impact de manière générale sur les lois de Zipf impliquées dans une image. Pour le vérifier nous pouvons extraire la répartition des motifs et les courbes de Zipf associées à plusieurs images chacune successivement laissée en couleur puis passée en niveaux de gris. Nous allons essayer d'étayer notre observation par un choix d'images variées. Nous pourrions ainsi valider ce constat empiriquement. Mais avant, nous pouvons déjà nous attendre à ne pas être surpris, puisque nous savons que notre méthode est globale et que finalement, comme nous segmentons notre espace colorimétrique en un même nombre de classes, on devrait dans la plupart des cas trouver des courbes très proches les unes des autres quand elles sont issues de la même image, en couleur puis en niveaux de gris. C'est ce que représente la figure 5.2.p. Expérimentalement nous n'avons pas rencontré d'image où la couleur apportait une information notable, mais il est facile de construire des images de synthèses pour lesquelles cette affirmation devient fausse.

Image	k-means couleur	k-means niveaux de gris
		
		
		
		

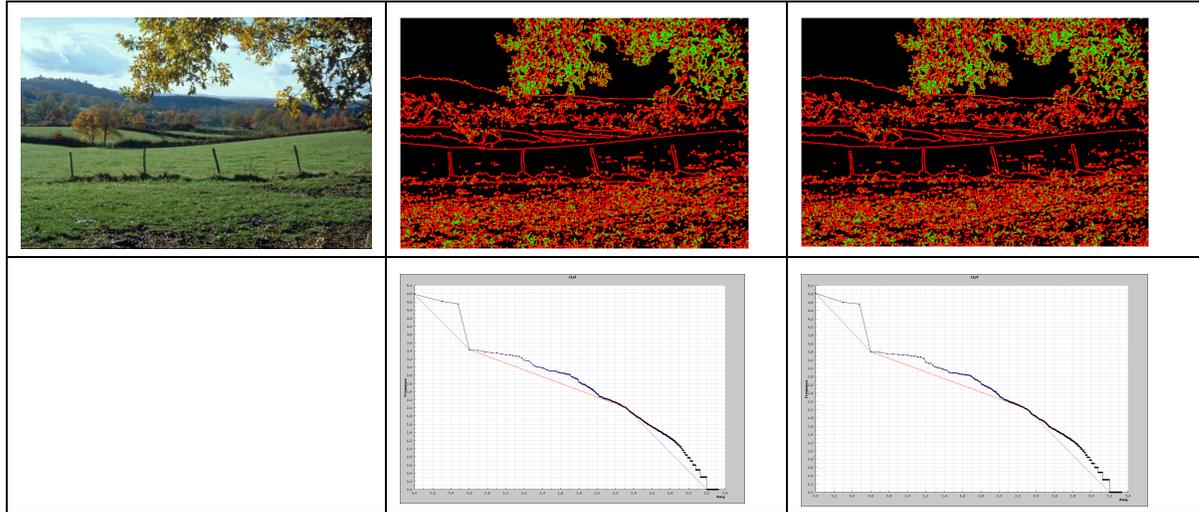


Fig 5.2.p : Exemples d'extraction de couches et de courbes de 3-means d'images en niveaux de gris et en couleur

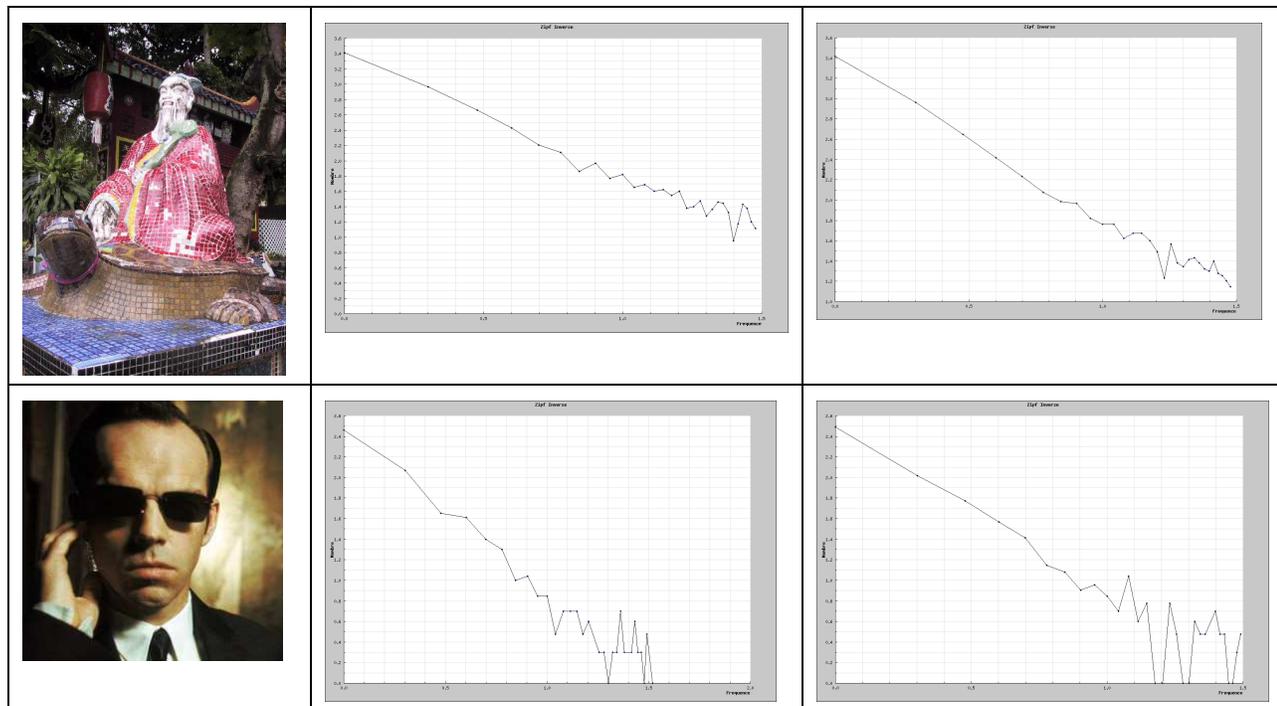
La figure 5.2.p nous montre que la loi de Zipf est un modèle réaliste aussi bien en niveau de gris qu'en couleur et que de plus à nombre de classes de couleurs égal, les courbes et même les couches extraites en couleur ou en noir et blanc sont proches l'une de l'autre. Le même phénomène que décrit précédemment touche les images en couleur, c'est-à-dire que pour être efficace la loi de Zipf nécessite un nombre de motifs différents possible ni trop grand, ni trop petit. Ce nombre semble surtout dépendre de la taille de l'image et non du type d'information associé au pixel, c'est pourquoi on remarque que les courbes de Zipf extraites de la même image mais une fois en couleur et une fois en noir et blanc sont extrêmement proches. Ce résultat n'est pas si étonnant et l'on comprend bien que la méthode des k-means appliquée sur une image couleur transformée en niveau de gris ou cette même méthode appliquée directement sur l'image couleur donnent des résultats presque identiques, surtout lorsque le niveau de quantification est inférieur à 256.

La méthode de classification des niveaux de couleur par les K-means et le codage basé sur celle-ci donnent des résultats étonnants proches d'une segmentation en trois couches de notre image. Cette segmentation en 3 couches ne dépend pas du nombre de classes des K-means mais simplement de la nature des courbes extraites qui présentent 3 parties significatives. Elle dépend aussi de la qualité de l'extraction des zones linéaires dans le graphe. Ce résultat n'est pas si étonnant que cela et l'observation de trois parties différentes dans une courbe de Zipf avait déjà été constatée dans d'autres études sur la loi de Zipf, appliquée sur des données monodimensionnelles. En effet des auteurs cités dans la section concernant la loi de Zipf section 3.2 ont déjà relevé ce phénomène [NEW05].

Il semblerait entre autre que pour être significative la loi de Zipf nécessite un nombre de motif possible ni trop grand ni trop petit. Cette dernière remarque se justifie simplement par le fait que la construction d'une courbe de Zipf ne comportant que très peu de motif n'aurait pas d'intérêt puisque cela mènerait à une courbe peu significative dont les caractéristiques propres auraient été gommées par une classification trop importante. Les motifs de contours seraient absorbés par le fond et donc un grand nombre d'informations serait perdu. A contrario un nombre de motifs possibles différents trop élevé mènerait à une courbe s'approchant d'une ligne horizontale où si l'on pousse le raisonnement à son paroxysme chaque motif présent dans l'image aurait une fréquence de 1, c'est-à-dire, ne serait représenté qu'une fois pour une faible proportion des motifs possibles.

5.2.3 La quantification et la loi de Zipf inverse

Maintenant que nous avons pu constater comment est impliquée la loi de Zipf classique nous pouvons observer ce que donne l'application de la loi de Zipf inverse avec ce codage. Nous devrions quelque soit le codage obtenir une courbe de Zipf inverse très linéaire et pouvant être caractérisée par le coefficient directeur de sa droite de régression linéaire. Dans la figure ci-dessous nous pouvons voir les courbes extraites de différentes images avec une application directe des k-means ou une application après conversion de l'image en niveaux de gris.



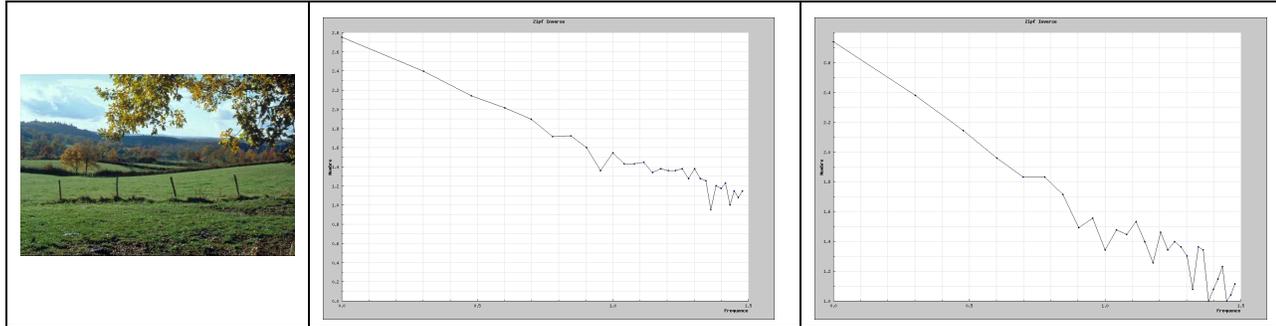


Fig 5.2.q : courbe de Zipf inverse extraite sur les images en couleur puis en niveaux de gris à l'aide d'un codage des 3-means.

Ce que nous constatons c'est que la loi de Zipf inverse est respectée que ce soit sur des images en niveaux de gris ou sur des images RGB. En effet les dix dernières fréquences forment un ensemble de points très rectiligne, caractéristique de la loi de Zipf inverse.

Ce que nous pouvons constater de plus, c'est que contrairement à ce que l'on a pu apercevoir avec les courbes extraites par la loi de Zipf classique avec un codage en 3-means, les courbes de Zipf inverse d'une même image en niveaux de gris et en RGB, ne se ressemblent pas. Ainsi si on caractérise ces courbes par le coefficient directeur de la droite de régression nous obtenons des résultats assez éloignés les uns des autres. Ce résultat n'est pas étonnant et il est normal qu'une zone uniforme en couleur le reste en niveaux de gris. Ainsi puisque le premier segment de la courbe de Zipf classique représente, comme nous l'avons vu dans les sections 5.2.1 et 5.2.2, les motifs issus de zones uniformes il est normal que ceux-ci ne varient que de epsilon entre le traitement fait sur une version couleur et celui fait sur une version en noir et blanc. De même que pour la seconde pente on peut voir, si on observe deux extractions d'une même image en couleur puis en noir et blanc, que la couche en rouge, celle dont les pixels appartiennent à des motifs du second segment de la courbe de Zipf classique, varie de façon très faible. Le résultat n'est toujours pas étonnant vu que ces motifs correspondent à des contours ou plutôt à des zones de transition entre les contours et les zones uniformes. C'est surtout en fin de courbe et donc sur les motifs les moins fréquents que les changements vont avoir lieu. Ainsi il devient évident que la loi de Zipf inverse, ne s'appliquant que sur les derniers motifs, soit touchée par une grande variation quand on l'extrait d'une image couleur ou d'une image en niveaux de gris.

Dans cette section nous avons pu voir la richesse qu'introduit le codage des motifs par une classification des niveaux de couleur. Plusieurs phénomènes ont put être modélisés et extraits des différentes courbes issues des images et ceci laisse penser que cette méthode pourrait donner de bons résultat si on utilisait les paramètres de ces courbes dans une application de recherche par le contenu.

5.3 Contraste et gradient

Dans cette partie nous allons essayer d'appliquer sur les images le codage dit des contrastes et des gradients. Cette méthode sera testée sur des images en niveau de gris ou sur des images couleur qui au préalable auront été transformées en niveaux de gris par une simple moyenne des trois canaux de l'histogramme RGB. Dans une première section nous nous intéresserons à la loi de Zipf classique puis dans une deuxième section nous verrons ce que donne la loi de Zipf inverse avec ce codage.

5.3.1 La loi de Zipf

La méthode que nous allons présenter s'intéresse plus au pixel et aux propriétés de son environnement que les autres méthodes.

La figure 5.3.a présente un exemple de la courbe de Zipf extraite, avec la méthode de codage des contrastes et des gradients, de l'image présentée en figure 5.2.a.

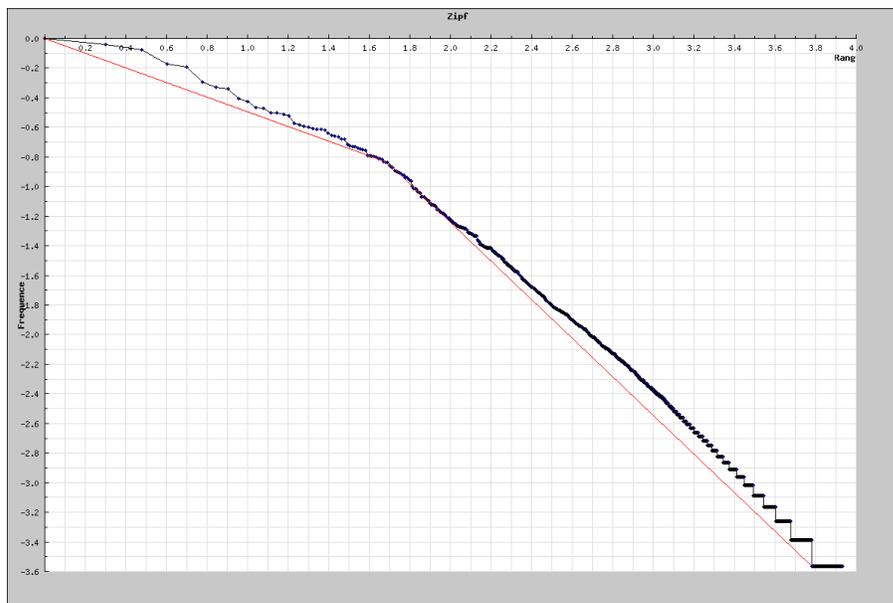


Fig 5.3.a : Courbe de Zipf de la figure 5.2.a avec le codage des contrastes.

Si nous observons cette courbe nous remarquons que la loi de Zipf ne s'applique pas et que nous voyons apparaître une courbe peu linéaire, pouvant à la rigueur être approximée par deux segments linéaires comme nous l'avons fait pour les k-means avec mais 3 segments.

Pour vérifier si cette découpe en deux parties a un sens nous pouvons extraire les pixels

composants par exemple la première partie de la courbe, le résultat est montré en figure 5.3.b.

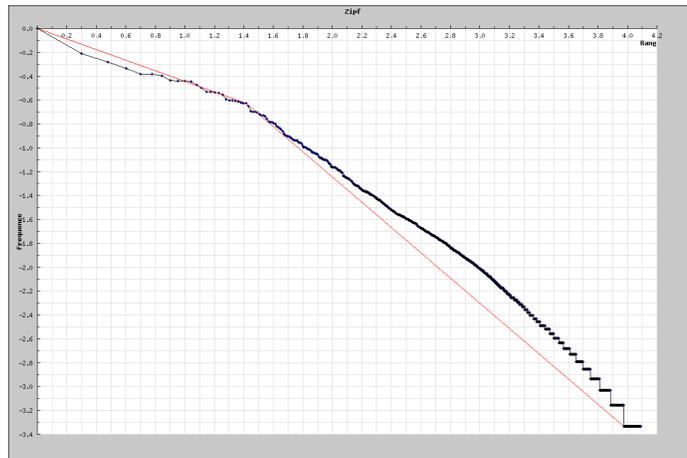


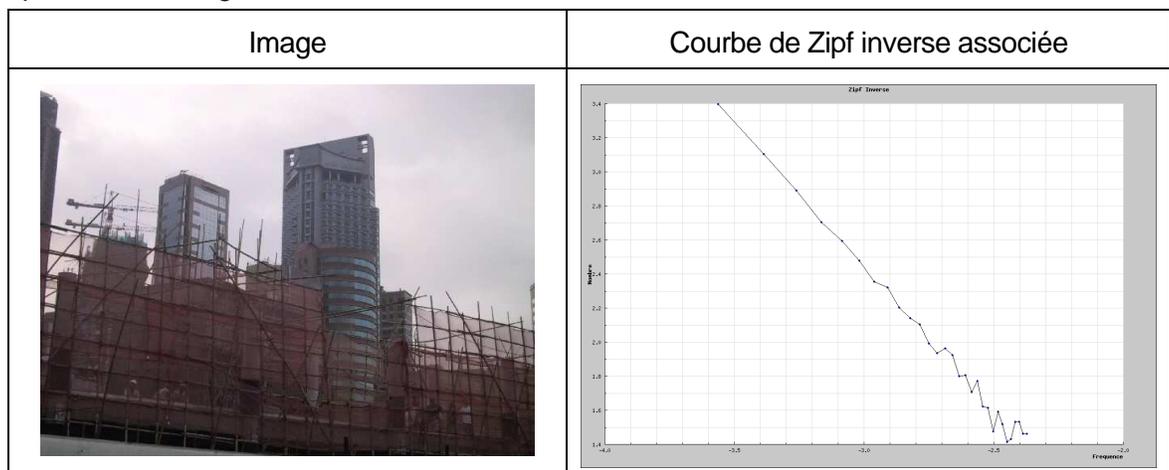
Fig 5.3.b : Couche 1 de l'image traitée par la méthode des contrastes et des gradients.

En observant la figure 5.3.b on se rend compte que la couche extraite à partir du premier segment de la courbe n'est pas aussi explicite que les couches que nous obtenons avec la méthode dite des k-means. Certes les pixels semblent correspondre à des pixels de surface mais tous ne sont pas extraits ce qui donne un effet d'inachevé à l'extraction. La méthode ne semble pas efficace et les courbes extraites ne semblent pas présenter toujours de parties très linéaires. De plus les motifs dans la dernière partie de la courbe se mélangent, c'est-à-dire que si on observe les différents motifs de la deuxième partie de la courbe ils alternent entre des motifs faisant partie d'une zone de surface homogène avec ceux représentant des coins et des contours.

On peut penser qu'une loi de Zipf unique n'est pas suffisante et que l'observation du graphe ne permet pas de mettre en évidence des zones linéaires correspondant à un mélange de lois qui s'adressent chacune à des populations de pixels ayant des portions différentes dans l'image, mais rassemblés dans un même phénomène.

5.3.2 La loi de Zipf inverse

Malgré le fait que la loi de Zipf semble peu s'appliquer avec cette méthode, il est tout de même intéressant de regarder ce que donne la loi de Zipf inverse. En effet nous avons vu que pour la méthode des k-means la loi de Zipf ne s'appliquait pas directement mais qu'en revanche la loi de Zipf inverse s'appliquait pleinement. Nous pouvons vérifier ceci pour notre codage.



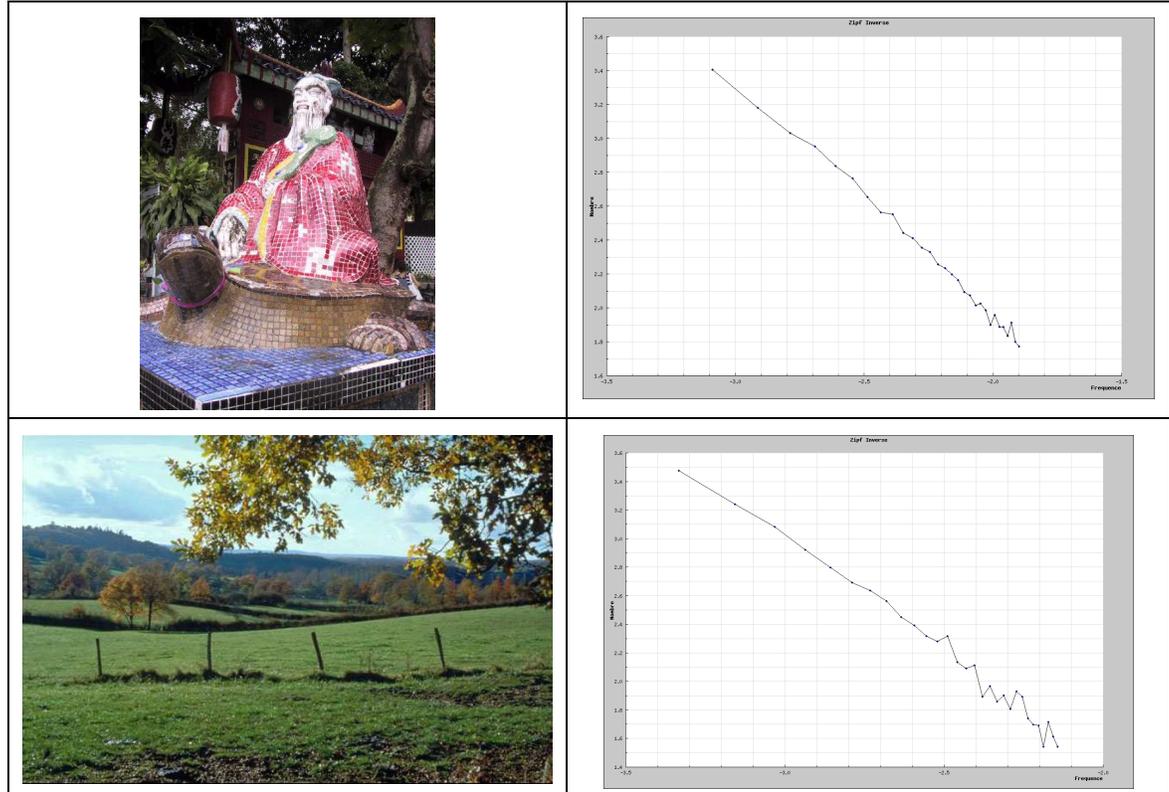


Fig 5.3.c : exemples d'images et de courbes de Zipf inverse associées

Contrairement à la loi de Zipf classique, la loi de Zipf inverse donne des courbes très rectilignes. On peut affirmer que la loi s'applique pleinement et que, de plus, elle donne des coefficients de droite de régression assez différents les uns des autres permettant de discriminer les différents types d'images.

Nous avons maintenant pu observer les différents phénomènes apparaissant suite aux différents codages quand on essaie de modéliser la distribution des motifs par la loi de Zipf pour les images. Nous allons dans le prochain chapitre nous intéresser à l'évaluation de ce modèle en considérant plusieurs applications pratiques et plusieurs bases d'images.

APPLICATIONS

6 Applications

Dans ce chapitre nous allons mettre en application certaines des différentes méthodes utilisées précédemment sur des problèmes réels. Nous allons essayer d'extraire des primitives, d'images provenant de plusieurs bases de données pour pouvoir évaluer notre approche sur des cas concrets. En effet il semble important de pouvoir évaluer l'efficacité de la loi de Zipf à indexer des images. Tout au long de nos expérimentations nous pourrions essayer de trouver à partir de nos courbes de Zipf et de Zipf inverse quelles sont les valeurs caractérisant les images. Tout au long de nos recherches nous avons privilégié une analyse globale et donc statistique plutôt que structurelle et locale. Nous avons remarqué que la loi de Zipf et son caractère global en font un outil simple et peu alambiqué. C'est pourquoi dans nos expérimentations de recherche d'images par le contenu nous nous attacherons toujours à utiliser les outils mathématiques et statistiques les plus simples. Il semble naturel que pour comparer les différentes courbes de Zipf ou de Zipf inverse de chacune des images nous utilisions des calculs de distance reposant sur la distance euclidienne entre point ou même de Hamming entre les différentes primitives extraites de l'application des lois de Zipf. Ou encore que pour classifier une image on utilise un simple algorithme des k plus proches voisins. Même si cette dernière remarque semble plus philosophique que scientifique elle est quand même appuyée par la volonté de tester l'efficacité de la loi de Zipf sur les images et non celle des outils nécessaires à son évaluation.

Ce chapitre est organisé en trois parties, nous nous attacherons dans une première section à classifier et à rechercher des images de lettrines provenant du Centre d'Etudes Supérieures de la Renaissance de l'université de Tours qui est depuis 1992 une équipe

de chercheurs associée au CNRS, puis nous évaluerons ensuite l'efficacité de notre méthode sur des images de manuscrit dits « des humanistes » et enfin dans une dernière partie nous utiliserons notre moteur de recherche avec des images en couleur.

6.1 Les lettrines

Pour pouvoir avoir une idée précise de l'efficacité de notre méthode il faut trouver une base de données que l'on pourrait qualifier d'objective, on entend par objective une base contenant des données dont les images peuvent facilement être étiquetées comme ressemblantes ou non l'une à l'autre. C'est dans ce cadre que la base de lettrines du CESR nous a paru un choix judicieux pour évaluer notre méthode. En effet ces images de lettrines et leurs proximités vont pouvoir être évaluées par une comparaison des courbes de Zipf mais surtout les résultats obtenus vont pouvoir être validés par les experts qui pourront affirmer ou infirmer notre méthode. Cette façon de valider les choses s'apparente à une véritable relation entre maître d'œuvre et maître d'ouvrage. Le CESR étant l'instigateur du projet de traitement des lettrines et nous, chargés de trouver les meilleures solutions techniques pour réaliser le projet. Ce contexte va nous permettre de trouver la forme de codage pour nos lois de Zipf et de Zipf inverse répondant de façon optimum au problème de recherche qu'a le CESR.

Si l'on s'en tient à la définition du dictionnaire, une lettrine est une lettre majuscule commençant un paragraphe ou un texte ayant une hauteur supérieure à celle de la ligne. Celle-ci est souvent ornée d'enluminures, terme issu du latin illuminare. C'est en fait à cause d'une erreur de vocabulaire entre le terme illuminare et miniature, venant du latin minimum que la lettrine est devenue plus qu'une simple lettre posée sur une série d'entrelacs et s'est transformée au cours des siècles en la représentation d'une scène en miniature.

On peut classer les lettrines en plusieurs groupes, en fonction de leur décoration, ainsi nous pouvons en définir quatre :

- ✠ Les lettrines rubriquées du latin ruber signifiant rouge, elles représentent les premières lettrines apparues dans les manuscrits, leur rôle n'était pas au début décoratif mais structurel. Elles permettaient au lecteur de rapidement repérer la structure d'un document. Leur nom vient du fait qu'au début elles étaient peintes en rouge et étaient très épurées voire même sans aucune ornementation.
- ✠ Les lettrines ornées, ou dites aussi lettres cadre, sont quant à elles plus travaillées et sont composées d'une lettre majuscule placée sur des entrelacs, des plantes, des animaux ou même un personnage sans pour autant représenter une scène proprement dit.

- ✦ Les lettrines synthétiques sont quant à elles les seules où la lettre n'est pas représentée réellement. C'est en fait les protagonistes de la scène qui, par leur agencement, font apparaître ou suggèrent la lettre.
- ✦ Les lettrines historiées représentent une scène, de la vie quotidienne ou encore narrative dessinée dans les espaces laissés libres par la lettre.

La figure 6.1.a ci-dessous illustre les différents types de lettrines.

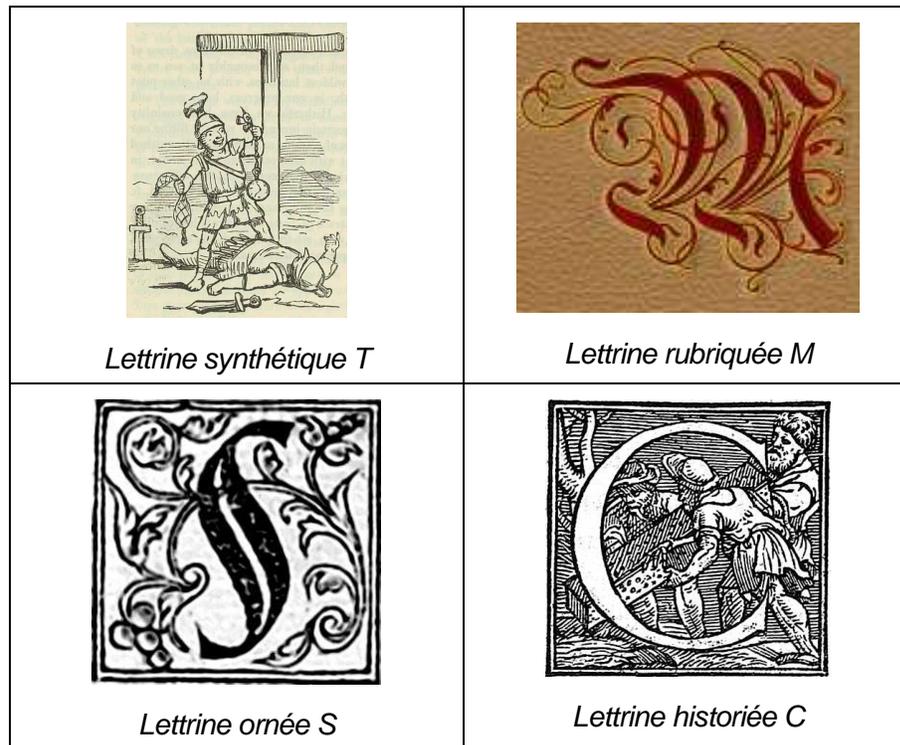


Fig 6.1.a : Exemple des différents styles de lettrine

On se rend compte qu'après n'avoir été qu'un simple outil structurant un manuscrit, la lettrine est devenue une illustration de plus en plus porteuse de sens. En effet la lettrine représente souvent une scène ayant un rapport avec le contenu du texte qu'elle enluminure. Plus encore, si au début elle fut peinte à la main et donc peu aisée à reproduire fidèlement, avec l'arrivée de l'imprimerie la lettrine devint finalement un petit tampon de bois trempé dans l'encre puis posé sur le papier. Cette technique de reproduction de lettrine va amener avec elle une somme d'informations historiques cachées. En effet les copistes pour des raisons évidentes d'économie furent amenés petit à petit à s'échanger et utiliser les tampons plusieurs fois. Ainsi on peut conclure qu'un ouvrage contient des informations dans le texte, mais aussi dans sa composition

physique. Il devient intéressant d'étudier les différentes parties le composant, comme sa couverture ou le papier de ses pages mais aussi on peut, en s'intéressant par exemple aux lettrines, retracer son histoire. Non seulement les jeux de lettrines s'échangent de copiste en copiste parcourant des distances qui peuvent être importantes mais elles se copient aussi et on peut constater que pas mal d'ouvrages sont des plagia d'autres. Plus encore des écoles d'enluminures se créent et donc donnent une certaine griffe à leurs productions. On peut encore pousser le raisonnement plus loin car avec le temps et l'utilisation un même tampon finit par produire des résultats sensiblement différents. Le tampon s'use et les parties saillantes finissent par s'aplatir. Ainsi en analysant cette usure on pourrait déterminer si deux copies d'un même ouvrage ont été réalisées à la même époque ou s'il s'agit d'une réédition.

Il semble que les lettrines aient, en fonction de leurs natures, de l'enlumineur qui les a réalisées, des caractéristiques communes. Dans un même ouvrage deux lettrines ne portent pas obligatoirement la même majuscule et ne représentent pas la même scène mais ont été réalisées par la même personne et illustrent une même histoire.

Ainsi on peut définir ce que l'on appelle le style d'une lettrine, en effet même si deux lettrines ne représentent pas la même lettre, elles peuvent être illustrées par des scènes composées des mêmes personnages dans des situations différentes. Le type de végétation ou d'entrelacs sur lequel est posée la lettre peut être sensiblement proche. La figure 6.1.b illustre ce concept.

Style	Exemple 1	Exemple 2	Exemple 3
Style A (Ornée)			
Style B (Ornée)			

<p>Style C (Historiée)</p>			
<p>Style D (Identique ou presque)</p>			
<p>Autres</p>			

Fig 6.1.b : Exemples de style de lettrines

Ces exemples illustrent parfaitement la notion de style, on voit qu'il n'y a pas de correspondance entre les lettrines d'un même style mais qu'une ressemblance générale ou plutôt globale existe. Le plus subtil étant le style D qui ne présente que des S mais venant de deux tampons différents et ne présentant pas exactement les mêmes enluminures.

Les lettrines que nous présentons viennent comme nous l'avons précisé de la base de données du CESR et ont été extraites à l'aide du logiciel Agora. Celui-ci a été conçu par l'équipe Reconnaissance des Formes et Analyse d'Images du Laboratoire d'Informatique de l'université de Tours. Il permet d'extraire les différentes parties d'un document ancien numérisé. Ainsi il peut séparer dans une page les bandeaux des lettrines, les lettrines du texte, etc. Il fonctionne à partir de scénarii élaborés par les experts du document. Il permet donc une extraction des lettrines sur une masse de document mais cette extraction de masse rend nécessaire l'indexation des images extraites. En effet étiqueter à la main les centaines de lettrines extraites se révélerait un travail titanesque.

Malheureusement, très peu de recherches ont été menées sur les images particulières que sont les lettrines, nous pouvons citer les études menées par le laboratoire de La Rochelle qui s'attache à indexer des lettrines. Leur méthode se base sur une segmentation de la lettrine en 3 couches, une première correspondant aux parties de texture, puis une autre composée des surfaces homogènes et la dernière qui correspond

aux contours. Une fois la lettrine segmentée, ils utilisent plusieurs méthodes de comparaison de graphe comme les arbres couvrants minimum, composés par les relations entre les centres de gravité des différentes zones de chacune des couches. L'efficacité de cette méthode est bonne et de nombreux résultats pertinents ont été obtenus [UTT05].

Pour tester notre méthode nous allons séparer notre base de lettrines comportant plus de 500 lettrines en deux. Une première partie sera une base dite d'apprentissage où les lettrines seront étiquetées en fonction de trois styles illustrés en figure 6.1.c. Le style 1 comporte 165 images de lettrines, le style 2 en comporte 20 et le style 3 en comporte 95.

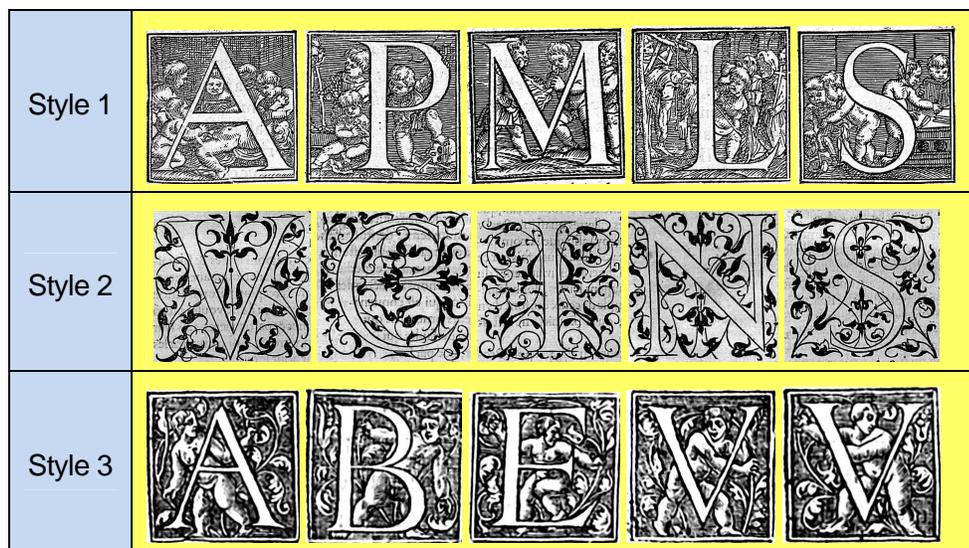


Fig 6.1.c : Exemple des trois styles utilisés par notre application

Nous allons extraire de toutes les lettrines de notre base les trois pentes de Zipf obtenues avec le codage par k-means. En effet la méthode des k-means est la méthode donnant des courbes présentant un sens entendez que la courbe se décompose en parties linéaires logiques segmentant l'image en trois couches. C'est dans cette perspective que nous allons utiliser la méthode dite des classes adaptatives ou k-means pour évaluer l'efficacité de la loi de Zipf. Celle-ci est bien adaptée au contexte de l'image et permet d'extraire 3 primitives ou plus d'une image. De plus ce codage permet d'allier à la méthode purement globale qu'est la loi de Zipf une note structurelle. Nous déciderons donc de caractériser une image de lettrine, par trois valeurs qui seront les coefficients directeurs des trois droites de régression, approximant les trois parties linéaires de la courbe de Zipf extraite de l'image.

Notre application va pouvoir se décomposer en trois, en effet trois possibilités s'offrent à

nous, on peut :

- A partir d'une image requête retrouver le style auquel elle appartient parmi ceux qui ont été préalablement définis.
- Toujours à partir d'une image requête déterminer les n images les plus ressemblantes se trouvant dans la base
- Etiqueter de façon automatique une masse de lettrines et déterminer les différents styles présents dans cet ensemble.

6.1.1 Moteur de recherche de lettrines

Nous avons, pour cette application, implémenté un outil totalement orienté web 2.0. Le code en devient non seulement portable mais aussi utilisable de n'importe où. La librairie utilisée est la librairie GD2 écrite en PHP.

Le principe de l'application est de trouver les n plus proches images de notre base de données de lettrines de la lettrine requête. Nous pouvons donner un aperçu des résultats obtenus avec notre application.

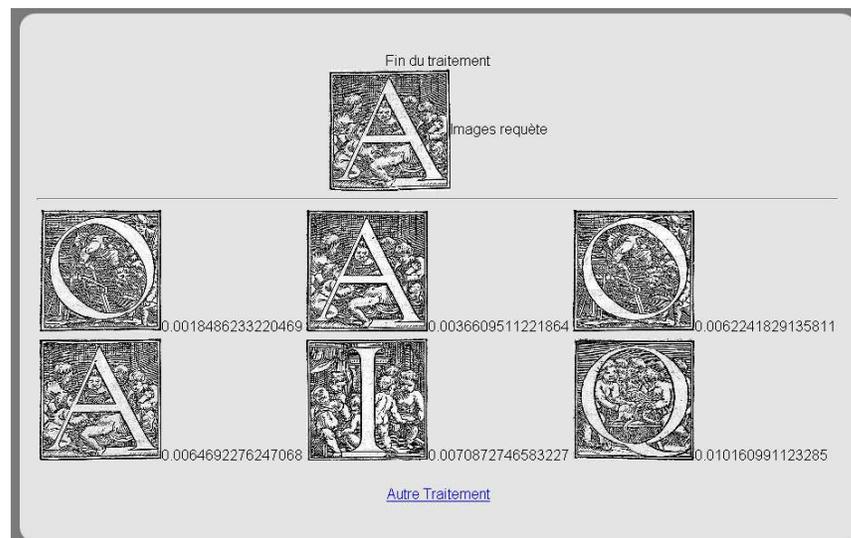


Fig 6.1.d : Résultat des 6 lettrines les plus proches du A de style 1

Le résultat semble plutôt satisfaisant puisque les 6 images les plus proches sont bien toutes du style 1. On peut faire de même avec nos autres styles en commençant par le style 2 comme montré en figure 6.1.e



Fig 6.1.e : Résultat des 6 lettrines les plus proches d'un I de style 2

Sur cet exemple il n'y a pas non plus d'ambiguïté les lettrines extraites sont bien des lettrines du style 2 comme l'image requête. On peut réaliser la même requête avec une image du style 3. Ceci est illustré en figure 6.1.f



Fig 6.1.f : Résultat des 6 lettrines les plus proches d'un I de style 3

La recherche d'images ressemblantes ne pose pas non plus de problème pour une lettrine du style 3.

On peut remarquer que la méthode ne tient pas compte de la lettre mais principalement du fond de celle-ci.

Il est aussi intéressant de tester les résultats que donne une lettrine du style un peu spécial présent dans notre base qui n'est composé que de lettrines portant la lettre S mais ne présentant pas toujours exactement le même ornement.

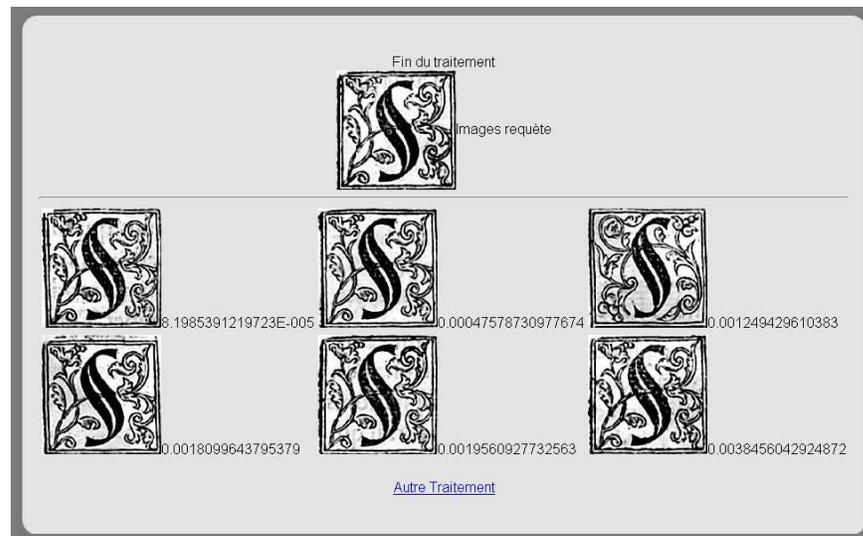


Fig 6.1.g : Résultat des 6 lettrines les plus proches d'un S de style spécial

Notre méthode semble très efficace pour retrouver des images ressemblant à une lettrine donnée. On valide aussi le fait que notre méthode est bien globale et qu'elle analyse l'image comme un agencement global de motifs. Ainsi notre moteur de recherche s'il retrouve des images de même style que l'image requête, il n'assure pas que les plus proches soient porteuses de la même lettre. Même pour le style présenté en figure 6.1.g, le S requête n'est pas strictement le même que le troisième plus proche.

6.1.2 Évaluation

Nous pouvons maintenant donner une évaluation quantitative de notre système en essayant de classer toutes les images de notre base dans un de nos trois styles ou de décider de ne pas l'affecter. Pour ce faire nous avons étiqueté avec l'aide des experts toutes les lettrines de notre base. Le but de notre application est de passer les lettrines dans notre algorithme afin de déterminer le style de chacune. Une fois le processus achevé il ne reste plus qu'à dénombrer les bons résultats et les mauvais. Nous allons évaluer avec cette application plusieurs niveaux de k-means. Nous allons utiliser la distance de Hamming pour évaluer les distances entre les trois pentes S associées à chaque lettrine I dans le graphe de Zipf.

$$\text{distance}(I, I') = \sum_{i=1}^3 |s_i - s'_i| \quad (611)$$

Les résultats sont présentés en figure 6.1.h. Pour déterminer le style d'une lettrine nous allons utiliser la méthode des n plus proches voisins avec n=1.

Codage	Style 1	Style 2	Style 3
Matrice 3x3 2means	88,48%	75%	79%
Matrice 3x3 3means	90,31%	80%	83,15%
Matrice 3x3 4means	85%	75%	80,1%
Matrice 3x3 9means	79,4%	65%	76,4%

Tab 6.1.1 : Résultat de reconnaissance de style avec le plus proche voisin

Les résultats du tableau 6.1.1 sont bons mais certaines erreurs persistent on va, dans un premier temps pouvoir améliorer ceux-ci en augmentant le nombre de plus proches voisins pris en compte. On remarque sans surprise que les résultats sont les meilleurs pour un niveau de k-means de trois. Nous avons déjà évoqué cette hypothèse dans la section 5.1.1.

Codage	Style 1	Style 2	Style 3
Matrice 3x3 2means	93,93%	80%	92,6%
Matrice 3x3 3means	98,8%	95%	94,7%
Matrice 3x3 4means	90,31%	90%	89,4%
Matrice 3x3 9means	85%	75%	78,9%

Tab 6.1.2 : Résultat de reconnaissance de style avec les 3 plus proches voisins

Si nous observons le tableau 6.1.2 nous constatons des résultats proches de 100% mais il reste encore des erreurs. Plusieurs indicateurs vont pouvoir nous aiguiller sur ces erreurs, ainsi si l'on regarde l'indice de confiance de nos n plus proches voisins on se rend compte que pour une bonne part des nouveaux bons résultats l'indice de confiance obtenu par application de la méthode des 3 plus proches voisins est souvent de 66,6% ce qui signifie que très souvent dans les trois lettrines les plus proches se trouve une lettrine d'un mauvais style. Il paraît intéressant d'observer les courbes de Zipf d'une image requête et de celle erronée mais néanmoins parmi les 3 plus proches images.

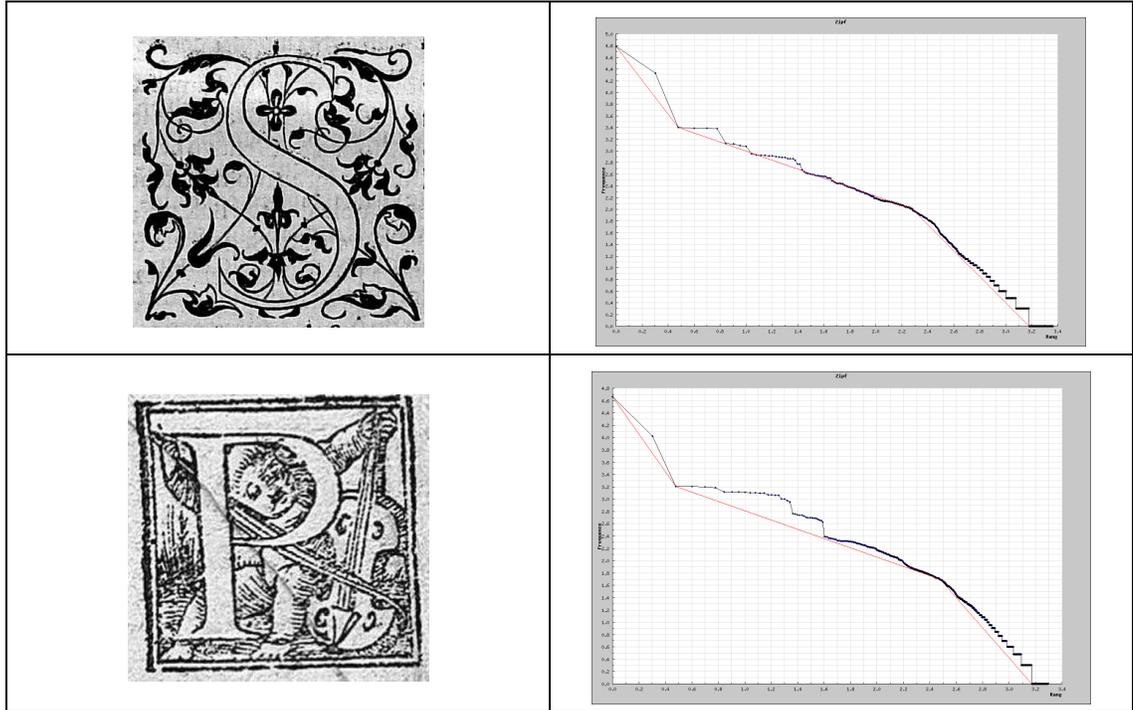


Fig 6.1.h : Exemple de lettrine requête avec une lettrine résultat erronée et leur courbe de Zipf respectives.

Si on observe attentivement la figure 6.1.h on se rend compte que les pentes sont très proches et qu'il est normal que cette image se soit retrouvée comme classée dans les plus proches de notre image requête. En revanche si les pentes sont proches on peut voir que les points de rupture entre les différentes parties ne sont pas les mêmes. On peut ainsi en déduire que cette information devrait être intégrée dans notre moteur de recherche. Ainsi une lettrine peut être caractérisée par sept valeurs les trois coefficients directeurs des droites de régression approximant les trois parties linéaires des courbes de Zipf et les trois points de rupture qui sont les abscisses des points limitant les trois parties de la courbe. Ceci est illustré en figure 6.1.i.

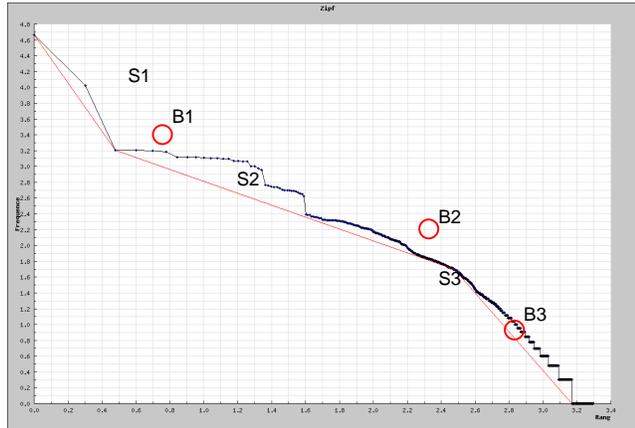


Image	S1	S2	S3	B1	B2	B3
-------	----	----	----	----	----	----

Fig 6.1.i : Illustration des primitives extraites d'une lettrine

Ainsi avec ces trois nouvelles primitives nous pouvons à nouveau relancer le processus de recherche de style. Les résultats sont pratiquement optimums comme le montrent les résultats dans le tableau 6.1.3. Résultats obtenus avec simplement le plus proche voisin.

Codage	Style 1	Style 2	Style 3
Matrice 3x3 3means	100%	93%	100%

Tab 6.1.3 : Résultat de reconnaissance de style avec le plus proche voisin

Les résultats ne sont pas optimaux mais si nous nous intéressons aux faux résultats nous découvrons que les erreurs viennent souvent non pas d'erreur de la méthode mais d'erreur d'étiquetage. En effet certaines lettrines ont été placées dans certains styles alors que si on les observe de plus près on remarque qu'elles ne sont pas complètement ressemblantes. On pourra parler d'image faussement d'un des styles présents dans la base d'image de lettrine. Ce phénomène est illustré en figure 6.1.j qui donne quelques exemples d'images de lettrines et des images de lettrines proches mais manifestement pas exactement du même style et pourtant étiquetées de la même façon.

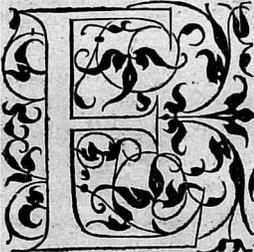
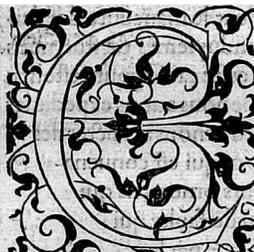
Lettrine bien étiquetée	Lettrine mal étiquetée
 <p data-bbox="787 724 876 756"><i>Style 2</i></p>	 <p data-bbox="1161 724 1323 756"><i>Faux style 2</i></p>
 <p data-bbox="787 1029 876 1060"><i>Style 2</i></p>	 <p data-bbox="1161 1029 1323 1060"><i>Faux style 2</i></p>
 <p data-bbox="779 1312 868 1344"><i>Style 1</i></p>	 <p data-bbox="1161 1312 1323 1344"><i>Faux Style 1</i></p>

Fig 6.1.j : Exemple de lettrines mal étiquetées

Une application indirecte que nous permise notre moteur de recherche de style a été la possibilité de trouver des erreurs d'étiquetage par l'humain. Cette application peut être très utile car on sait que lors de classement manuel d'un grand nombre d'éléments, les erreurs sont inévitables. Ainsi nous pouvons, grâce à notre méthode, valider ou non un classement manuel par une rapide vérification de cohérence ou trouver un intrus dans un ensemble d'image à priori de même style.

Si les résultats de notre moteur de recherche ont été améliorés par l'ajout de trois nouvelles primitives, on peut imaginer que ceux du moteur de recherche de lettrines l'ont été aussi. Il paraît essentiel d'observer l'impact qu'ont eu les points de rupture sur le

classement des images les plus similaires à une lettrine requête. Nous allons nous intéresser aux lettrines de style 1. La figure 6.1.k présente une série de tests de recherche de lettrines similaires.

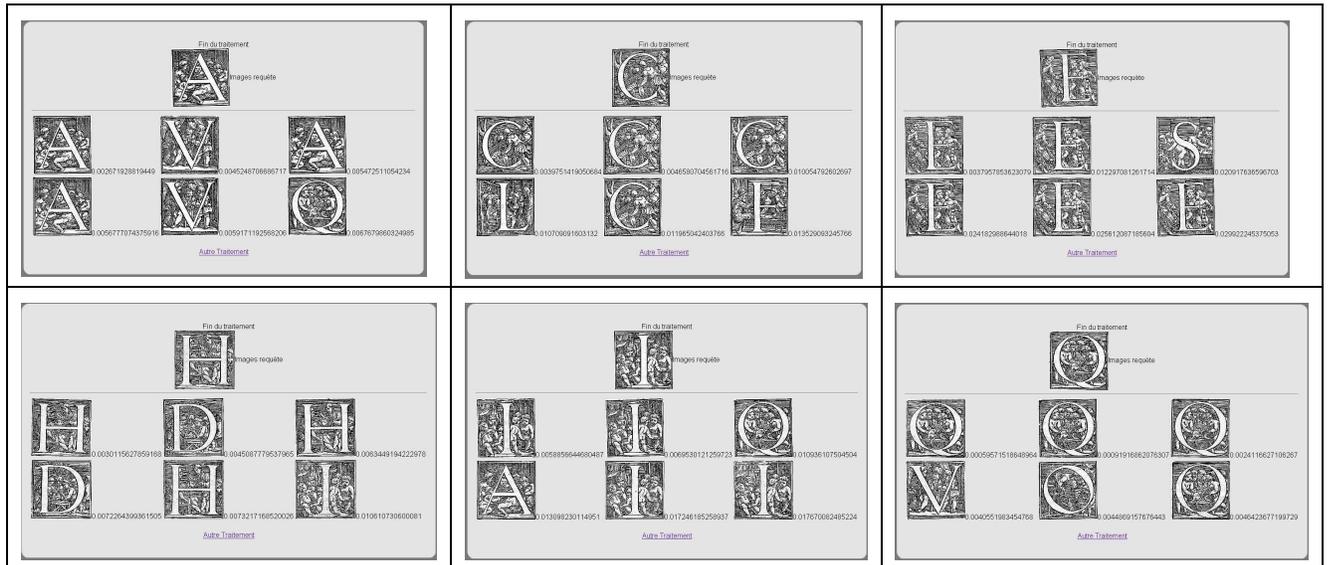


Fig 6.1.k : Résultat des 6 lettrines les plus proches d'une lettrine de style 1

Ce que l'on remarque sur les tests présentés en figure 6.1.k c'est que l'efficacité de notre moteur de recherche a été accrue. En effet nous remarquons que sur les six plus proches images trouvées toutes sont du style 1. Mais surtout ce qui est notable c'est que, contrairement à ce que l'on avait vu auparavant quand on ne tenait compte que des pentes pour effectuer notre recherche, les lettrines proches de notre image requête sont pour la plupart des lettrines portant la même lettre. Ce qui est encore plus remarquable c'est par exemple pour la lettrine portant un A des V sont aussi retrouvés et en réfléchissant un V est grossièrement un A à l'envers. Le cas du Q est aussi notable puisque l'on trouve aussi dans les six premières lettrines un O qui d'un point de vue structure n'est pas très éloigné. Les quelques lettrines erronées, entendez ne portant pas la bonne lettre sont là car leur arrière plan et donc la scène qu'elles représentent sont proches au niveau structure.

Si l'on s'interroge un peu sur la méthode on s'aperçoit finalement qu'à partir d'une méthode très globale on a réussi à définir un codage et des primitives issues de celui-ci qui donnent une vision plus structurée de la lettrine. Même si quelques erreurs subsistent dans notre moteur de recherche il n'en reste pas moins que les résultats sont plus qu'encourageants et vont plus loin que ceux d'une méthode générale qui discriminerait les images uniquement dans une vision ultra globale de celle-ci. Pour

étayer nos conclusions nous devons mesurer les résultats que donne une méthode classique d'indexation d'images en niveaux de gris. C'est ce que nous nous proposons de faire dans la section suivante.

6.1.3 Évaluation par les histogrammes

Nous avons choisi de comparer notre méthode à celle des histogrammes pour plusieurs raisons. La première est que cette méthode est très répandue et comme nous l'avons vu dans la section sur l'état de l'art de l'indexation d'images, celle-ci peut être qualifiée de classique et de maîtrisée. On entend par ces qualificatifs que le volume de recherche et de bons résultats obtenus avec cette méthode en font un pilier de l'indexation d'image. Une deuxième raison du choix de cette méthode est sa simplicité et sa nature globale, en effet on ne peut que comparer ce qui est comparable et il est clair que vu la qualification globale de la loi de Zipf il faut nous mesurer à une méthode similaire.

Nous procéderons de la même façon et sur la même base d'images que pour l'évaluation des k-means toujours avec deux applications : un moteur de recherche de lettrines similaire ainsi qu'un moteur de recherche de style.

La méthode va reposer sur l'extraction de l'histogramme de toutes les lettrines de notre base. Mais dans un souci de performances et surtout vu la nature ternaire de nos images de lettrines il est judicieux de regrouper les niveaux de gris en plusieurs classes ici nous utiliserons 5 classes. Une fois obtenus les histogrammes des différentes lettrines seront comparés à l'aide du coefficient de corrélation dont la formule est indiquée ci-dessous.

$$r_p = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \quad (612)$$

Pour calculer le coefficient de corrélation entre deux matrices $X(x_1, x_2, \dots, x_n)$ et $Y(y_1, y_2, \dots, y_n)$ on peut utiliser la formule suivante :

$$r_p = \frac{\sum_{i=1}^N (x_i - \bar{x}) \times (y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \times \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}} \quad (613)$$

Plus le coefficient se rapproche de 0 moins les histogrammes sont corrélés et plus il est proche de 1 plus elles sont corrélées et donc proches.

- Moteur de recherche de lettrines par les histogrammes

Nous allons tout comme pour les lettrines présenter ici des exemples de résultats donné

par la méthode des histogrammes dans la recherche de lettres similaires dans notre base. Nous considérerons comme précédemment les trois styles présents dans notre base ainsi que celui ne comportant que des lettres portant la lettre S.

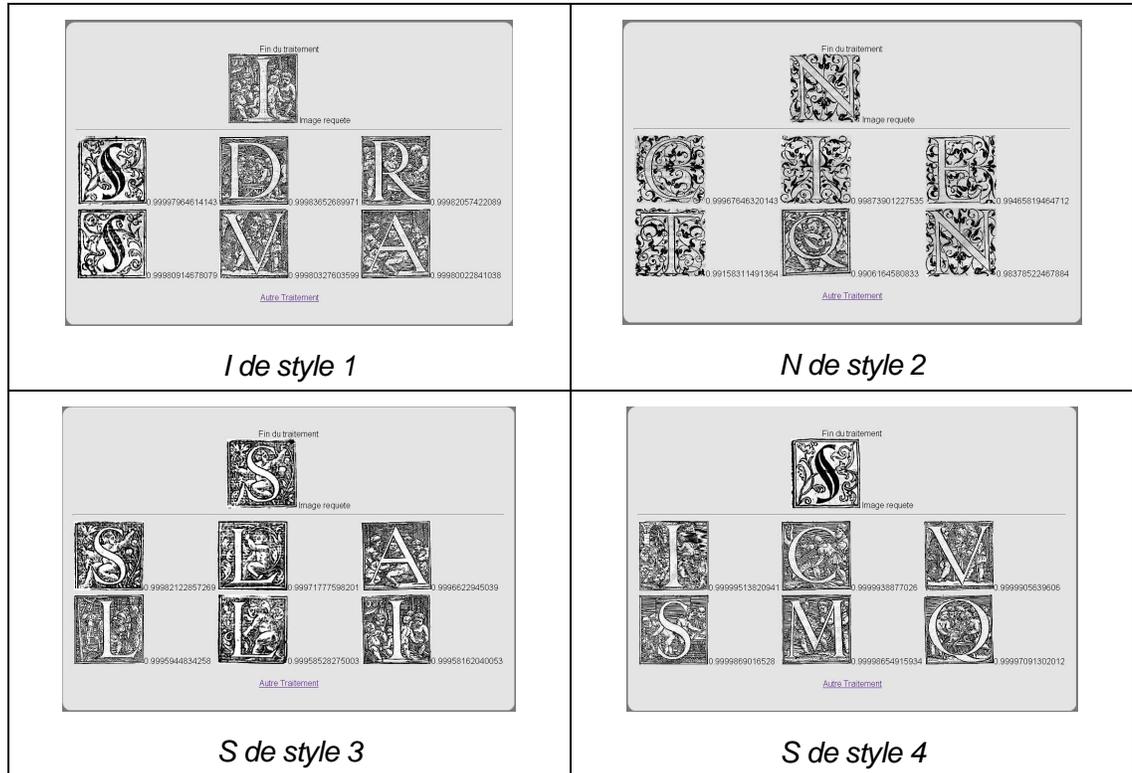


Fig 6.1.1 : 4 Résultats des 6 lettres les plus proches d'une lettre requête

Ce qui semble frappant sur ces exemples c'est que le moteur n'est pas aussi pertinent pour la recherche de style que celui utilisant le codage des k-means. Même si la majorité des lettres retrouvées excepté pour le style 4 sont bien du même style que l'image requête, il n'en reste pas moins que des erreurs apparaissent et de temps en temps celles-ci se trouvent en première position. Que dire du cas particulier du style 4 ne comportant que des S où les 6 images les plus similaires sont toutes d'un style différent.

- moteur de recherche de style

Toujours utilisant le protocole appliqué précédemment, nous allons essayer, à l'aide du calcul des histogrammes de retrouver le style d'une lettre requête à l'aide des n plus proches voisins. Nous allons évaluer les résultats corrects de notre recherche avec deux valeurs de n, 1 et 3. Les résultats sont donnés dans le tableau 6.1.4.

NPPV / Style	Style 1	Style 2	Style 3
N=1	80%	75%	85,26%
N=3	96,9%	85%	89,4%

Tab 6.1.4 : Résultats de reconnaissance de style avec les plus proches voisins

Les résultats sont bons mais le coefficient de confiance de nos NPPV sont faibles. Beaucoup d'erreurs sont commises et la méthode par histogrammes ne souffre pas la comparaison avec l'usage de la loi de Zipf. Ces résultats ne sont pas étonnants puisque les histogrammes ont tendance à perdre toute notion d'organisation structurée au sein de l'image alors que nous avons vu que le codage de la loi de Zipf par les k-means gardait une petite part de la structure de l'image. Il en découle que si la détermination du style peut se faire avec les histogrammes c'est surtout dans l'application dite du moteur de recherche de lettres similaires que l'écart se creuse réellement. En effet nous avons vu que l'algorithme des k-means arrivait souvent à placer dans les images les plus ressemblantes des lettres provenant non seulement du même style mais aussi portant le même caractère. De plus il n'a pas été trouvé de style pour lequel la méthode ne fonctionnait pas du tout ce qui n'est pas le cas pour les histogrammes puisque le style 4 doit avoir un taux de reconnaissance par les histogrammes proche de 0% quelle que soit la valeur des NPPV utilisés.

6.2 Les manuscrits

Afin d'évaluer d'une façon objective notre méthode et ne pas s'attacher à une base d'images à laquelle serait bien adaptée notre méthode. Nous nous proposons dans cette section de nous attaquer à un domaine qui est en pleine effervescence, qui est l'indexation de manuscrits. De nombreuses recherches sont en cours sur ce sujet et il pourrait être judicieux de tester notre méthode sur ces images de texte. Nous testerons notre méthode sur des manuscrits dits des humanistes. L'objectif est de pouvoir authentifier une écriture à partir de l'examen d'une missive non signée. En effet l'apparence de l'écriture est liée au contexte du scripteur. Ce contexte au sens large du terme est composé de l'éducation ainsi que l'aptitude qu'a une personne à plus ou moins bien maîtriser ses mains. On sait qu'un artiste peintre présentera une écriture plus déliée que celle du commun des mortels. On sait aussi que l'écriture évolue avec l'âge. Ainsi on peut affirmer que l'écriture est une caractéristique propre à chacun et peut permettre de savoir si un texte a bien été écrit par une personne ou non [SRI01]. L'étude de l'écriture des individus et surtout l'authentification d'un document n'est pas une tâche dénuée de

sens. En effet la graphologie est un outil essentiel de notre système d'investigation juridique et les experts du domaine sont souvent appelés à authentifier des écrits notamment pour des raisons soit financières, héritage ou autres, soit même pénale pour identifier l'auteur d'une lettre de chantage ou de menace.

6.2.1 L'indexation d'images de manuscrits

L'indexation d'images de manuscrits se décline en plusieurs groupes. Trois se détachent en fonction de la nature des manuscrits numérisés. En effet les différentes langues ne présentent pas les mêmes caractéristiques quant à la formation des lettres formant les mots. Ainsi on pourra séparer les manuscrits de langue asiatique, de ceux de langue arabe de ceux utilisant un alphabet occidental. Nous ne nous intéresserons ici qu'aux manuscrits occidentaux. Avec la nécessité de numériser les manuscrits anciens pour les sauvegarder est arrivée la nécessité de les indexer. Nous ne reviendrons pas sur le côté caduc d'une base d'images sans système d'indexation.

Nos images de documents anciens nous éloignent des applications classiques d'indexation de manuscrits que sont la validation de signature pour les chèques bancaires ou encore de reconnaissance d'adresses postales manuscrites en vue d'un tri automatique du courrier. Le but des méthodes va être ici soit de dater des documents, soit d'identifier le scripteur d'un document. Sur ce domaine bien précis on peut noter quelques recherches qui ont mené à des résultats efficaces. Ainsi Thierry Paquet du LITIS de l'université de Rouen utilise les champs markoviens pour vérifier si l'organisation spatiale des données respecte les caractéristiques de l'écriture de Flaubert [PAQ04]. Une autre orientation est d'utiliser une fonction d'auto corrélation pour mettre en avant la périodicité et la direction des textures. Cela mène à la construction d'une rose directionnelle qui permettra de caractériser l'écriture [ELA05]. Cette méthode part d'un simple constat, les auteurs affirment que contrairement aux textes imprimés les textes manuscrits subissent de grosses variations d'écartement non seulement entre les mots mais aussi entre les caractères eux mêmes.

Ces méthodes sont celles, que l'on pourrait qualifier de récentes et viennent s'ajouter aux méthodes plus classiques c'est-à-dire étant les fondations du domaine. Nous n'aurons pas ici la prétention de faire ici un état de l'art de toutes les méthodes d'identification de scripteur. Nous présenterons en revanche les méthodes se rapprochant de la nôtre qui sont dites, par styles. Ces méthodes extraient des primitives caractéristiques d'un scripteur sans tenir compte de la signification du texte. Deux types de méthodes, que l'on pourra qualifier de globales sont utilisées aujourd'hui :

Le premier type de méthodes [NIE07-BUL07] définit l'écriture comme une succession d'allographes eux-mêmes composés de coup de stylo nommés allotraits. Dans [CRE95-

SED98] la méthode utilise onze primitives, on y trouve l'épaisseur du trait, le corps du texte ainsi que l'espacement de l'écriture, et huit autres qui sont liées à l'orientation du texte.

La deuxième approche utilise la dimension fractale de Minkowski pour mesurer l'irrégularité des écritures [VIN95- TAN02- BUN01].

$$D(x) = \lim_{r \rightarrow 0} \left[2 - \frac{\ln A(X_r)}{\ln r} \right] \quad \text{Avec } A(X) \text{ l'aire d'un ensemble } X \text{ (621)}$$

A l'aide de cette dimension et de différentes dilations des ensembles X , on va aboutir à deux visions de l'écriture, une dite de près et une autre dite de loin. Cette première étape permet de distinguer une écriture très régulière type imprimée d'une écriture plus irrégulière type manuscrite. Des caractéristiques supplémentaires peuvent être ajoutées à cette méthode et surtout on a introduit l'utilisation de la compression fractale [VIN98]. Ainsi en compressant puis en décompressant une image et mesurant le rapport signal sur bruit entre image originale et celle décompressée on peut classer les écritures par style.

C'est dans la même optique que ces dernières méthodes que nous allons essayer d'évaluer notre méthode sur des images de documents manuscrits.

6.2.2 Identification de scripteur par la loi de Zipf

Nous allons appliquer notre méthode d'indexation d'images sur les manuscrits décrits en partie 6.2. Le principe reste le même que pour les lettrines sauf que au lieu de style nous avons maintenant affaire à des scripteurs. Ainsi notre base d'image est composée de manuscrits écrits par différents scripteur. Notre base contient au moins 3 manuscrits écrits par un même scripteur. Le but de notre évaluation est de retrouver à qu'el scripteur appartient un manuscrit que l'on propose à notre application.

Avant de montrer les résultats obtenus nous allons tout de même observer les courbes de Zipf extraites, par les différents codages, de nos images particulières que sont les manuscrits.

- Les rangs généraux

La figure 6.2.a montre des exemples de courbes et leurs manuscrits associés.

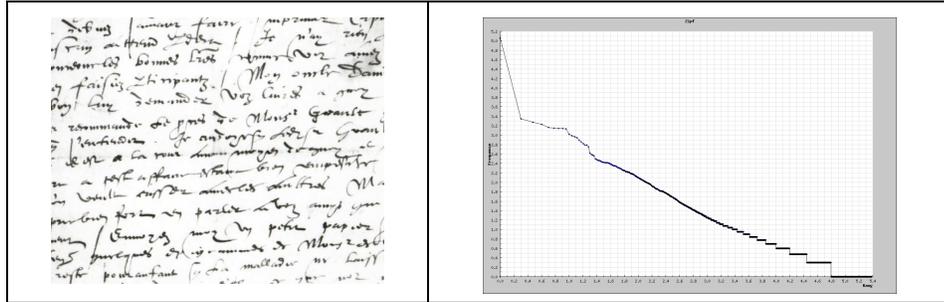


Fig 6.2.a : Exemple d'une image de manuscrit et de sa courbe de Zipf associée extraite par la méthode des rangs généraux

Nous observons un résultat sans surprise avec une partie de la courbe très linéaire avec comme nous l'avons remarqué dans la partie 5.1.1 un motif représentant l'uniforme très présent et ensuite une courbe pouvant être approximée par le coefficient linéaire de la droite de régression obtenu à partir du deuxième motif. L'effet d'escalier en fin de courbe est toujours présent. Seul petit bémol à ajouter à notre observation est l'apparition d'une partie de courbe un peu moins linéaire entre les motifs de rang 2 à environ 20. Ce phénomène en fonction des manuscrits peut être plus ou moins prononcé comme l'illustre la figure 6.2.b.

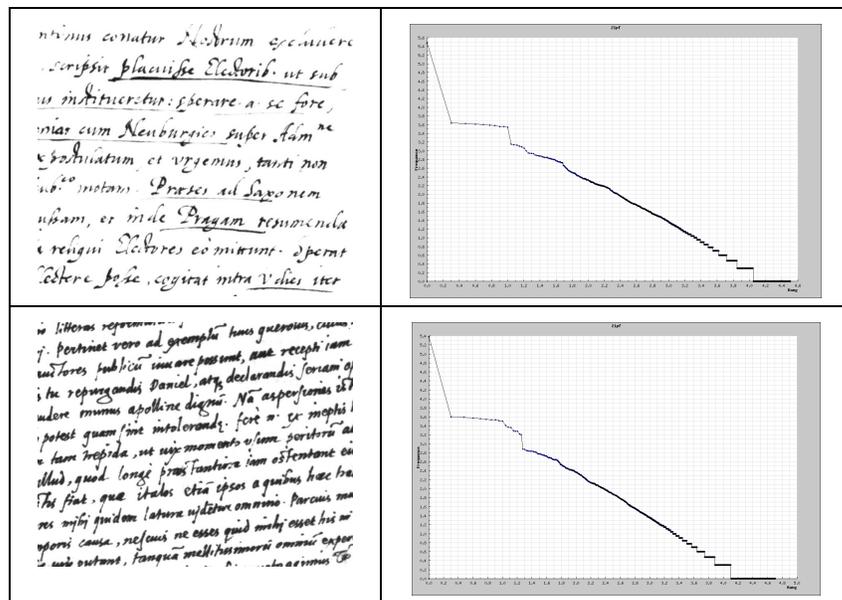


Fig 6.2.b : Exemples d'images de manuscrits et de leurs courbes de Zipf associées extraites par la méthode des rangs généraux

Ce phénomène amène au sein de notre méthode un petit peu de structure puisque si l'on

observe la figure 6.2.c correspondant à la représentation des motifs de rang 2 à 9 on se rend compte que ceux-ci ont une structure très particulière.

Motif	Rang	Fréquence	Motif	Rang	Fréquence
1 1 1	2	4287	1 0 1	6	3847
1 0 1					
1 1 1					
0 1 1	3	4222	1 1 1	7	3805
1 1 1					
1 1 1					
1 1 1	4	4107	1 1 0	8	3623
1 1 1					
1 1 0					
1 1 1	5	4010	1 1 1	9	3579
1 1 0					
1 1 1					

Fig 6.2.c : Motifs de rang 2 à 9 de la figure 6.2.a

La structure des motifs correspondants à la partie de notre courbe qui ne respecte pas la linéarité générale de celle-ci est particulière. Ces motifs comportent un pixel foncé entouré par des pixels plus clairs et de niveaux de gris identiques. On retrouve donc une organisation non aléatoire des motifs de nos images de manuscrits.

- Les contrastes

La figure 6.2.d donne une illustration de l'application de la méthode dite des contrastes documentée dans la section 5.3.1. Cette application se fera en respectant scrupuleusement ce qui a été décrit plus haut.

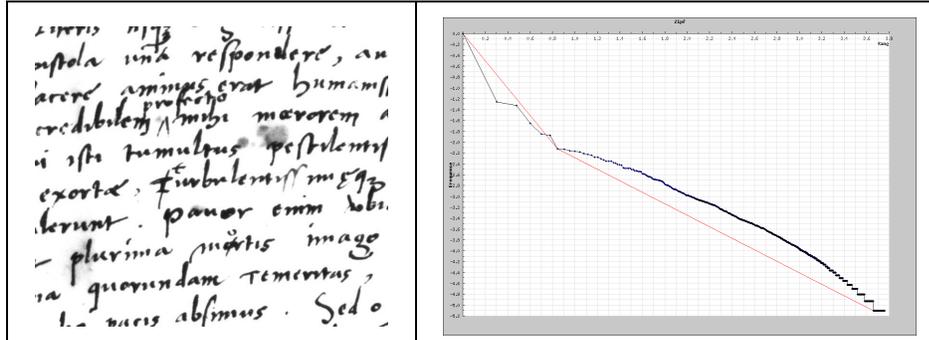


Fig 6.2.d : Exemple d'image d'un manuscrit et de sa courbe de Zipf associée extraite par la méthode des contrastes

La méthode débouche sur une courbe qui présente deux parties linéaires. Ainsi la première qui représente les motifs les plus fréquents est composée de motifs que l'on pourrait qualifier de halo c'est-à-dire la zone un peu grise entre le noir de l'écriture et le blanc du papier. A nouveau nous voyons que cette méthode ne respecte pas scrupuleusement la loi de Zipf mais met en évidence un aspect structural de nos images de manuscrits.

- Les classes

Nous ne pouvons pas traiter les manuscrits sans tester sur eux la méthode qui donne de très bons résultats sur d'autres parties des documents anciens que sont les lettrines. L'efficacité relevée dans notre application de moteur de recherche de lettrines nous oblige à effectuer les mêmes tests sur cette nouvelle base de données.

La nature des images nous portera à conserver tout au long de nos tests le même niveau de k-means que pour les lettrines. Cette valeur ayant les mêmes justifications pour les manuscrits que pour les lettrines, nous ne nous attarderons pas plus sur ce point.

C'est donc avec un motif de 3x3 pixels et une classification en 3 classes adaptives que nous allons présenter les courbes de Zipf issues des images de manuscrits. Un exemple de résultat est donné en figure 6.2.e

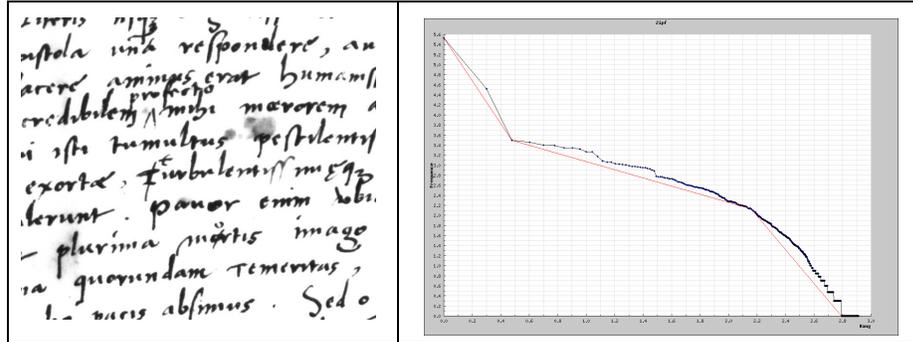


Fig 6.2.e : Exemple d'image d'un manuscrit et de sa courbe de Zipf associée extraite par la méthode des k-means

Les résultats obtenus ne sont pas étonnants et nous nous attendions à retrouver la même forme de graphe que pour les lettrines. Nous pouvons observer aussi la répartition géographique de nos motifs en affichant les trois couches composant l'image de manuscrit.

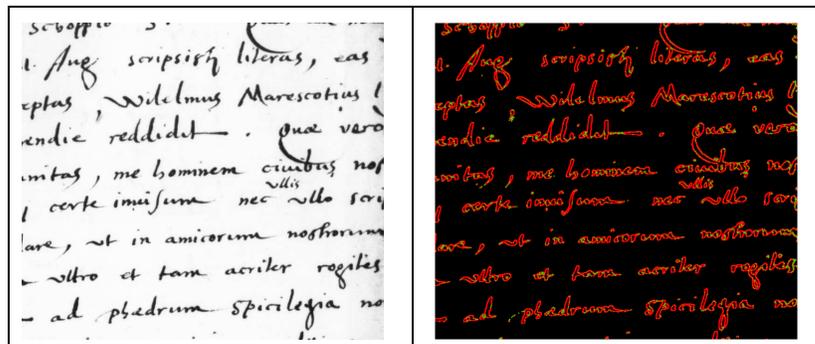


Fig 6.2.f : Extraction des trois couches correspondant aux trois parties de la courbe de Zipf

Les résultats sont bons et nous obtenons encore une approche un peu plus structurée de notre méthode statistique. Il paraît donc naturel d'effectuer nos évaluations avec cette méthode, ce qui nous permettra, de plus, de valider celle-ci pour les documents anciens.

6.2.3 Évaluation

Nous nous proposons d'évaluer ici le nombre de bonnes reconnaissances de scripteur que donne une distance de Hamming entre les primitives extraites d'une image requête et toutes les primitives des images déjà indexées dans notre base de données. Pour rappel, les primitives utilisées ici sont les coefficients directeurs des trois droites de régression approximant la courbe ainsi que les trois points dits de rupture situés aux extrémités de chacune des parties de la courbe de Zipf. Le résultat d'une requête dans

notre moteur de recherche de scripteur donne des résultats assez bons mais non optimum. On peut voir une illustration de nos tests en figure 6.2.g



Fig 6.2.g : exemple de résultats d'une recherche de manuscrit proches

Les résultats présentés montrent que les deux premiers manuscrits les plus proches de notre image requête sont bien du même scripteur que celle-ci. En revanche le troisième est erroné. Si nous testons et évaluons notre méthode sur la base entière nous obtenons les résultats présentés dans le tableau 6.2.1. Ces résultats ont été extraits à l'aide d'un algorithme de N plus proches voisins. Nous avons cantonné N à deux valeurs soit 1 soit 3, ceci se justifiant par la présence en général de 3 exemples de manuscrit d'un même scripteur.

N=	1	3
Taux	55%	62%

Tab 6.2.1 : Évaluation de la reconnaissance de scripteur par 6 primitives

Les résultats ne sont pas aussi performants que pour les lettrines et peuvent déconcerter mais il faut analyser les données pour mieux comprendre ce faible taux de reconnaissance. En effet, souvent les erreurs ne sont souvent évaluables qu'avec l'étiquetage et à l'œil il est très difficile de savoir si l'outil se trompe ou non. Notre méthode étant globale elle est plus encline à déterminer des styles généraux plutôt que d'analyser finement un style particulier. Il semblerait que la nature de notre base comporte trop de styles d'écriture proches les uns des autres. On avait déjà dû ajouter à notre évaluation des primitives nouvelles pour la rendre plus efficace, il semble donc nécessaire à nouveau d'extraire une nouvelle valeur caractéristique de nos images de manuscrits pour mieux les discriminer.

6.2.4 La loi de Zipf inverse et les manuscrits

L'idée d'appliquer la loi de Zipf inverse sur la base des manuscrits nous est venue

naturellement et se justifie par ce que l'on avait énoncé dans la section 5.2.3 c'est-à-dire que la loi de Zipf inverse est plus à même d'extraire des informations plus fines que la loi de Zipf classique. Nous l'avons vu avec les représentations couleur et noir et blanc d'une même image et leurs courbes de Zipf respectives associées, qui si elles n'étaient pas totalement identiques étaient vraiment très proches l'une de l'autre, ne différant que sur quelques motifs qu'on qualifiera de fin de courbe. En revanche les courbes de Zipf inverse de ces mêmes images présentaient des pentes totalement différentes. Nous pourrions conclure que la loi de Zipf inverse peut-être utilisée quand on cherche à indexer des choses très proches et qu'une analyse plus fine, entendez moins globale, est nécessaire. Il semble que la base de manuscrits que nous possédons présente des différences entre styles très complexes à observer et que la loi de Zipf inverse a du mal à extraire, il se pourrait que la loi de Zipf inverse puisse être plus efficace. Nous donnons un exemple d'une courbe de Zipf inverse associées à une de nos images de manuscrit, en figure 6.2.2.

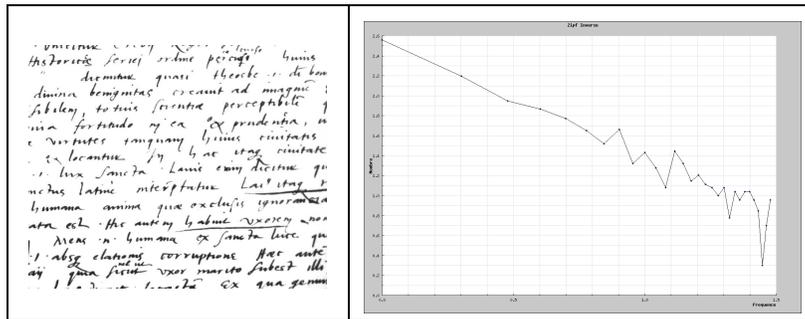


Fig 6.2.2 : Exemple de courbe de Zipf inverse extraite d'une image de manuscrit

Nous constatons que le manuscrit ne déroge pas à la règle qui veut que l'application de la loi de Zipf inverse mène à une courbe dont la première partie est très linéaire. Il paraît intéressant d'introduire dans les primitives caractérisant nos images de manuscrits le coefficient directeur de la droite de régression de la partie linéaire de la courbe de Zipf inverse.

Cette nouvelle primitive introduite, il ne reste plus qu'à relancer l'évaluation, et regarder si les résultats sont meilleurs ou non. Nous donnons en figure 6.2.3 une illustration de notre moteur de recherche de manuscrits.

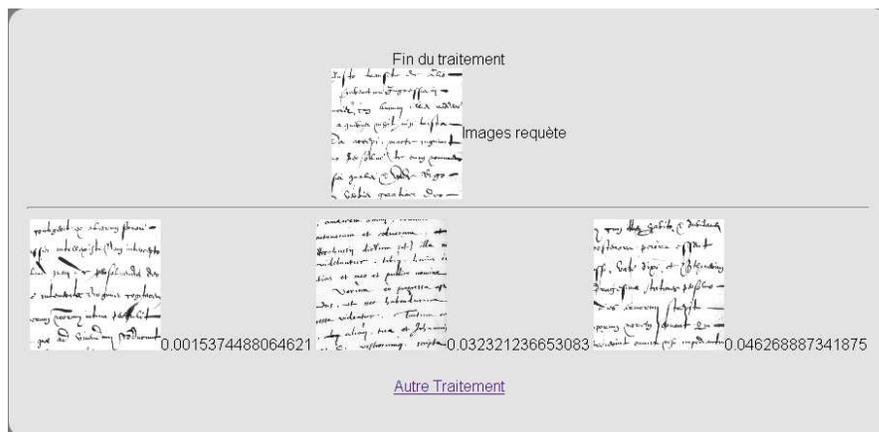


Fig 6.2.3 : 3 manuscrits les plus proches de l'image requête

L'évaluation est plutôt bonne et l'on se rend compte que deux manuscrits trouvés sont du même scripteur que le manuscrit de l'image requête. De plus si l'on observe le manuscrit retrouvé mais n'étant pas de la main du même scripteur il est tout de même très proche stylistiquement de l'image requête. Il est assez remarquable que pour évaluer notre méthode l'on soit obligé de se référer aux étiquetages car souvent à l'œil nu nous sommes incapables de discriminer les mauvais résultats des bons. Notre moteur donne comme résultat d'une requête souvent des manuscrits d'un même scripteur mais surtout des manuscrits présentant la même orientation le même espace d'interligne et la même épaisseur de trait.

Nous allons évaluer maintenant les résultats que nous obtenons avec notre application d'identification de scripteurs. Les résultats sont présentés dans le tableau 6.2.2.

N=	1	3
Taux	65%	81%

Tab 6.2.2 : Évaluation de l'apport de la loi de Zipf inverse sur l'identification de scripteurs

Les résultats sont bons mais loin de ceux que nous obtenons sur les lettrines. Ceci peut s'expliquer par la trop grande similarité entre les styles d'écritures présents dans notre base d'images. Ce que nous notions lors de l'évaluation de notre première application reste vrai ici. La difficulté, à différencier les manuscrits, présente à l'œil nu le reste avec notre méthode globale. En revanche, sur l'identification fine de scripteur notre méthode peut être utilisée pour effectuer une sorte de pré-classification permettant, sur un volume immense d'images de manuscrit, d'éliminer en les classant par style d'écriture une très grande majorité de textes similaires. Il ne restera à étudier qu'un nombre plus réduit de documents. On remarque d'ailleurs que c'est ce que l'on réalise avec la base de lettrines est en fait une pré-classification de celles-ci.

6.3 Les images couleur

Dans les sections précédentes nous nous sommes cantonné à évaluer notre méthode sur des images en niveaux de gris. Ceci nous coupe d'une partie non négligeable des images produites aujourd'hui par les différents outils de numérisation et de capture d'images équipant non seulement les professionnels mais aussi les particuliers.

Aux difficultés techniques introduites par le nouvel espace colorimétrique tridimensionnel s'ajoute la nécessité de trouver une méthode d'évaluation aussi objective que celle de l'identification de scripteurs ou de classification de style des lettres. Nous avons rassemblé en une base d'images 400 images en couleur de plusieurs types. On y trouve aussi bien des photos d'identité, que de paysages naturels ou urbains ainsi que des textures. La figure 6.3.a présente un échantillon représentatif de la base d'image utilisée.



Fig 6.3.a : Échantillons de la base de données d'images couleur

La base d'images semble être pour le moins éclectique et va nous permettre de tester notre méthode sur un panel d'images couleur très différentes.

6.3.1 Évaluation

Nous allons extraire les primitives que nous avons utilisées lors de l'évaluation sur les manuscrits c'est-à-dire que nous aurons sept valeurs caractéristiques pour chacune des images. Les trois coefficients directeurs de la droite de régression approximant les trois parties de nos graphes de Zipf, ainsi que les points extrêmes de chacune des trois parties et enfin le coefficient directeur de la droite de régression approximant le graphe de Zipf inverse. Nous avons montré dans la section 5.2.2 que l'utilisation d'un algorithme de k-means avec $k=3$ donnait des résultats plus probants quant à l'extraction de nos courbes de Zipf.

Nous pouvons illustrer de plusieurs exemples ce que donne une application de recherche d'image par le contenu sur notre base d'image en utilisant notre méthode. C'est ce que présente la figure 6.3.b.

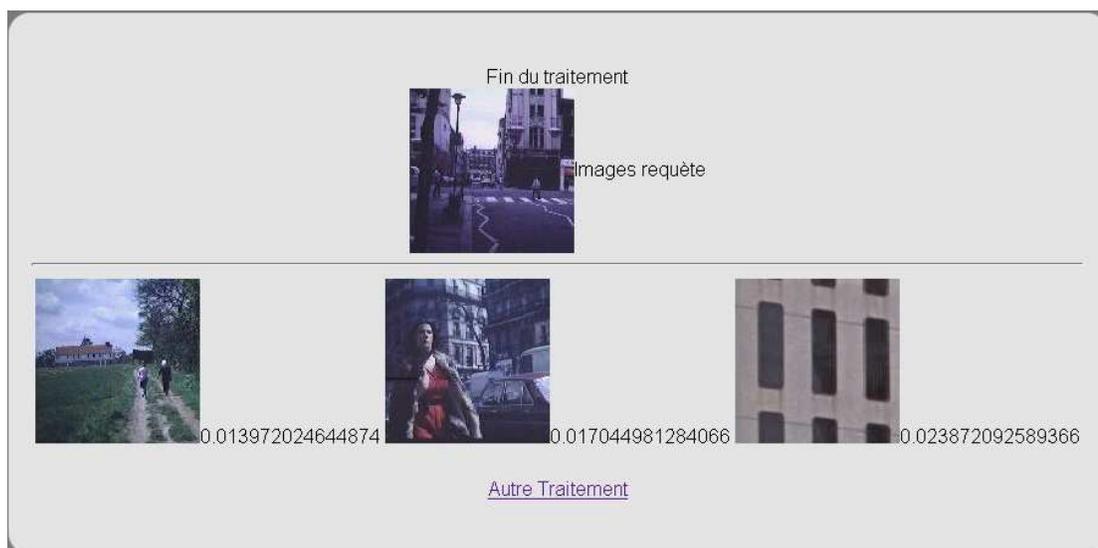


Fig 6.3.b : exemple de résultat de notre moteur de recherche

L'exemple de la figure 6.3.b montre les résultats obtenus de similarité entre une image requête représentant une rue avec des voitures et un piéton et ses trois plus proches voisins dans la base. L'image la plus proche est un couple de piétons marchant sur un chemin tracé en direction d'une construction. La deuxième image la plus proche est une femme marchant dans les rues d'une ville. La troisième est une partie d'un bâtiment. Ces images sont proches les unes des autres toutes dans une « certaine mesure ». En effet il est difficile de mesurer la similarité entre deux images naturelles sans avis objectif. Notre base est très difficile à étiqueter et nous ne pouvons nous en remettre, comme pour les évaluations précédentes, à l'avis d'experts.

Nous allons continuer à illustrer une recherche dans notre base en figure 6.3.c



Fig 6.3.c : exemples de résultats de notre moteur de recherche

Notre méthode n'est pas parfaite et de nombreux résultats posent encore quelques problèmes. En revanche la méthode est encourageante et les similarités entre les images retrouvées sont souvent grandes.

6.3.2 Conclusion

Il semble que la méthode soit encourageante mais que le domaine laisse encore pas mal de champ pour de nouvelles recherches. On pourrait dans un premier temps s'intéresser à l'extraction automatique des parties linéaires des graphes de Zipf que l'on pourrait

améliorer. Le choix des primitives caractérisant une image peut être aussi un sujet de discussion. En effet une sélection de primitives peut être envisagée. Nous savons ce que chacune des primitives illustre et donc nous pouvons jouer sur celles-ci en fonction de ce que l'on cherche à extraire comme similarité au sein des images. En effet on sait par exemple que la première partie du graphe représente les zones uniformes de l'image, les autres étant les contours. Il semble qu'une analyse du contexte soit une aide précieuse pour l'efficacité de notre méthode.

CONCLUSION & PERSPECTIVES

7 Conclusion & perspectives

Nous avons développé une méthode d'indexation d'images, générale, basée sur l'utilisation de deux lois puissance, les lois de Zipf et de Zipf inverse. Nous avons montré comment adapter au contexte de l'image, ces lois habituellement utilisées dans des domaines qualifiés de monodimensionnels. Il a fallu, pour mettre en oeuvre ces lois, se basant sur les fréquences d'apparition de motif au sein d'un phénomène, transposer la notion de motif au domaine de l'image. Nous avons donc présenté plusieurs possibilités de codage de ces motifs pour des images en niveaux de gris ou en couleur. Nous avons vu que les codages présentaient tous des caractères propres quant à l'extraction des courbes de Zipf mais qu'un phénomène récurant se propageait transversalement entre eux. En effet il est possible quel que soit le codage d'approximer la courbe de Zipf, associée à un codage, par un ou plusieurs segments de droite, du fait de l'apparition d'une ou plusieurs parties linéaires dans la courbe. Cette constatation nous a permis de vérifier que la loi de Zipf pouvait bien être appliquée à l'indexation d'images et permettait d'extraire des valeurs caractéristiques représentant une image.

Du fait de la nature globale des lois puissance nous pouvons qualifier notre méthode d'indexation d'images, de méthode globale et ce malgré le fait qu'avec certains codages, notamment celui utilisant l'algorithme des k-means, nous avons pu mettre en exergue quelques phénomènes structurels. De toute façon l'exposant des lois est un indice sur la structure qui apparait. Ces phénomènes structurels ne gardent pas une information spatiale de l'image mais la segmente en trois couches de pixels représentant chacune un type de structure présent dans l'image. Si l'image correspond à un mélange de phénomènes, l'ordre de grandeur des fréquences impliquées permet dans certains cas d'extraire les éléments du mélange. Nous avons ainsi pu constater que l'image pouvait être représentée par un premier ensemble de pixels correspondant aux zones uniformes

de celle-ci, un deuxième ensemble représentant les contours extérieurs, c'est-à-dire ceux se trouvant dans les zones de transition entre la texture et les contours à proprement parler, représentés eux par le troisième ensemble de pixels. L'efficacité de cette segmentation, en trois couches, de notre image a pu être mise en évidence avec différents types d'images, aussi bien en niveaux de gris qu'en couleur. Nous avons pu constater aussi que pour que la loi de Zipf soit optimum dans le domaine de l'image il fallait trouver un nombre de motifs possibles que l'on qualifiera d'équilibré. Entendez que ce nombre ne doit être ni trop grand ni trop faible et que la qualité des résultats décroît avec l'éloignement du nombre de motifs possibles, présenté par un codage, de cette valeur équilibre. Ainsi nous avons pu constater que pour la méthode utilisant l'algorithme des k-means l'équilibre était atteint lorsque l'on attribuait une valeur de trois à k. Ce qui peut-être noté aussi c'est que cette valeur semble être la même, que l'on soit sur une indexation d'images en niveaux de gris ou d'images couleur.

Nous avons pu évaluer notre méthode sur des images de documents anciens et nous avons pu montrer que les primitives extraites des images représentant des choses différentes pouvaient être prise en compte ou non en fonction du type de similarité que l'on cherchait à représenter. Ainsi les pentes des segments linéaires du graphe de Zipf d'une image donnent une information sur la structure de l'image et celle-ci varie très peu que l'image soit en couleur ou en niveau de gris. Nous avons pu constater que les courbes de Zipf obtenues sur des images couleur étaient très proches de celles obtenues sur les mêmes images mais en niveaux de gris, mais que leurs courbes de Zipf inverse étaient très dissemblables. Il est donc nécessaire pour discriminer des images en niveaux de gris de leurs équivalent en couleur, d'utiliser la pente de la droite de régression approximant la courbe de Zipf inverse de celles-ci.

Tout au long de nos applications nous avons introduit des primitives nouvelles et vu les résultats que cela donnait. Nous pouvons affirmer que sur les documents anciens la loi de Zipf est une méthode très efficace pour classer par style ceux-ci. Nous avons pu le constater avec les lettrines ainsi qu'avec les manuscrits. Nous avons eu aussi des résultats encourageants sur la base d'images en couleur malgré le fait que l'évaluation soit plus complexe à réaliser.

Cette dernière réflexion nous mène directement aux diverses perspectives que laisse entrevoir cette thèse. En effet, il est clair qu'il pourrait être utile de mesurer l'impact qu'on les diverses primitives extraites et ce tous codages confondus sur les similarités retrouvées entre les images. On pourrait s'attacher à définir des panels de primitives à utiliser en fonction des attentes de l'utilisateur en matière de ressemblance.

Une étude plus approfondie sur chacune des couches que nous obtenons avec le codage utilisant l'algorithme des k-means pourrait se révéler très enrichissante. Il est à

noter que d'autres travaux utilisent une segmentation en couches sémantiques pour l'analyse et l'interprétation des images de documents [TOM02-SUR05].

Nous pouvons aussi imaginer que sur certains domaines notre méthode pourrait s'avérer très efficace lors de pré traitements pour réaliser une pré classification d'images.

On peut aussi essayer de définir de nouveaux codages pour les motifs et s'attacher d'avantage aux problèmes particuliers qu'apportent avec elles les images en couleur. Ainsi on peut s'interroger sur l'apport qu'aurait l'utilisation d'autres espaces colorimétrique que l'espace RGB classique. En effet nous avons trois dimensions qui jouent des rôles symétriques alors que l'espace TSL permettrait de différencier les traitements associés à chaque dimension.

Enfin en perspective on pourrait aussi essayer d'adapter notre méthode à d'autres domaines et appliquer la loi de Zipf et la loi de Zipf inverse aux vidéos par exemple. En essayant de ne tenir compte non plus seulement de l'organisation spatiale des différentes images composant une séquence mais aussi de l'organisation temporelle en considérant des motifs à cheval sur différentes frames.

PUBLICATIONS DE L'AUTEUR

Rudolf Pareti, Nicole Vincent, "Global Discrimination of Graphic Styles", Graphics Recognition ten years review and future perspectives, LNCS 3926, Liu and Lladós eds, 2005

Rudolf Pareti, Nicole Vincent, "Reconnaissance de scripteur par une loi puissance", Colloque International Francophone sur l'Écrit et le Document, (CIFED) Fribourg, Suisse Francophone, 2006

Rudolf Pareti, Nicole Vincent, "Global Method Based on Pattern Occurrences for Writer Identification", 10th International Workshop on Frontiers in Handwriting Recognition (IWFHR), La Baule, 2006

Rudolf Pareti, Nicole Vincent, "Ancient Initial Letters Indexing", 18th International Conference on Pattern Recognition, (ICPR), 2006 Hong-Kong

BIBLIOGRAPHIE

- [ADA00] L.A. Adamic B.A. Huberman, "The nature of markets in the World Wide Web", Quarterly Journal of Electronic Commerce 1, 512, 2000
- [ADA01] S. Adam, J.-M. Ogier, C. Cariou, R. Mullot, J. Gardes, Y. Lecourtier, "Utilisation de la Transformée de Fourier-Mellin pour la reconnaissance de formes multi-orientées et multi-échelles : Application à l'analyse automatique de documents techniques", Traitement du Signal, 18:17-33, 2001
- [ALP84] Alperovich, Gershon, 1984. "The size distribution of cities: On the empirical validity of the rank-size rule," Journal of Urban Economics, Elsevier, vol. 16(2), pages 232-239, September
- [AZE87] R. Azencott, "Image analysis and Markov fields", ICIAM, pp53–61, Paris, 1987
- [BUL07] M. Bulacu, L. R. B. Schomaker, "Text-independent writer identification and verification using textural and allographic features", IEEE Trans on Pattern Analysis and Machine Intelligence (PAMI), Special Issue, Biometrics: Progress and directions, April, 29(4), p. 701-717, 2007
- [BEG89] A. Beghdadi, A. Negrata, "Contrast enhancement technique based on local detection of edges", Computer Vision, Graphics and Image Processing, Academic Press Professional Inc. 1989, pp. 162-174, vol. 46, May 1989, San Diego, USA
- [BEN38] F. Benford, "The law of anomalous numbers", Proceedings of the American Philosophical Society, 1938, p. 551
- [BI97] D. Bi, "Segmentation d'images basée sur les statistiques de rangs de niveaux de gris", Thèse de doctorat, Université de Tours, 1997
- [BLA07] B. Blasius, R. Tönjes, A. Bandrivskyy, "Zipf law in popularity distribution of chess openings", 2007 eprint arXiv:0704.2711.
- [BLO99] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, J. Malik, "Blobworld: A system for region-based image indexing and retrieval", Third Int. Conf. on Visual Information Systems, June 1999
- [BOU01] N. Boujema, J. Fauqueur, M. Ferecatu, F. Fleuret, V. Gouet-Brunet, B. Le Saux, H. Sahbi, "Ikona : Interactive specific and generic image retrieval", International Workshop on Multimedia Content-Based Indexing and Retrieval, 2001

- [BRO05] E. L. Van Den Boek, P. M. F. Kister, L. G. Vuurpijl, "Content-based image retrieval benchmarking : utilizing color categories and color distrutions", *Journal of Imaging Science and Technology*, 49(3), p. 293-301, 2005
- [BUN01] U. -V. Marti, R. Messerli, H. Bunke, "Writer indentification using text line based feature", 6th International Conference on Document Analysis and Recognition (ICDAR'01), 2001
- [CAR02] Y. Caron, P. Makris, N. Vincent, "A Method for Detecting Artificial Objects in Natural Environments", *icpr*, p. 10600, 16th International Conference on Pattern Recognition (ICPR'02) - Volume 1, 2002
- [CAR04] Y. Caron, "Contribution de la loi de Zipf à l'Analyse d'images", thèse de doctorat Université de Tours, 2004
- [CHO03] C.-W. Chong, P. Raveendran, R. Mukundan, "A comparative analysis of algorithms for fast Computation of Zernike moment", *Pattern Recognition*, 36:731-742, 2003
- [COH97] A. Cohen, R. N. Mantegna, S. Havlin, "Numerical analysis of word frequencies in artificial and natural language texts", *Fractals* 5(1), p. 95-104, 1997
- [CRE95] J.-P. Crettez, "A set of handwriting families : style recognition", *Conférence on Document Analysis and Recognition (ICDAR'95)*, Montréal, Canada, p. 489-494, 1995
- [DEL03] E. Dellandrea, "Analyse de signaux video et sonores : application à l'étude de signaux médicaux", Thèse de doctorat de l'université de Tours, 2003
- [DEM96] L. Dempsey, S. L. Weibel, "The Warwick Metadata Workshop: a framework for the deployment of resource description", *D-Lib Magazine*, July/August 1996
- [DER00] S. Derrode, M. Daoudi, F. Ghorbel, "Invariant content based image retrieval using a complete set of Fourier-Mellin descriptors", *Computer Vision and Image Understanding*, pages 57-78, 2000
- [DOC01] C. Doctorow, "Metacrap: Putting the torch to seven straw-men of the meta-utopia", 2001
- [DUB89] R. C. Dubes A. K. Jain, "Random fields models in image analysis", *Journal of applied statistics*, vol. 16, no. 2, 1989, pp. 131-164.
- [EBE02] H. Ebel, L.-I. Mielsch, S. Bornholdt, "Scale-free topology of e-mail networks", *Phys.Rev.E* 66, 035103, 2002
- [ELA05] A. El Abed, V. Eglin, F. Lebourgeois, H. Emptoz, "Frequencies decomposition and partial similarities retrieval for patrimonial handwriting documents compression", In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 996-1000, 2005.
- [ENN95] F. Ennesser, G. Medioni, "Finding Waldo, or Focus of Attention Using Local Color Information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 805-809, Aug., 1995
- [FAR96] N. Faraj, R. Godin, R. Missaoui, S. David, P. Plante, "Analyse d'une méthode d'indexation automatique basée sur une analyse syntaxique de texte", *Canadian Journal of Information and Library Science/Revue l'information et de bibliothÉconomie*, 1996, 21(1), 1-21

- [FAU04] J. Fauqueur, "Contributions pour la Recherche d'Images par Composantes Visuelles", Thèse de doctorat de l'Université de Saint-Quentin en Yvelines, 2004
- [FAU06] A. Faure, "Classification et commande par réseaux de neurons", Hermes, 2006, 216 p
- [FIO94a] N. George, J. Evans, N. Fiori, J. Davidoff, B. Renault, "Apport des potentiels évoqués cérébraux à l'étude de l'effet de "supériorité des visages"". *Psychologie Française*, 39(4), 313-326, 1994
- [FIO94b] N. George, J. Evans, N. Fiori, J. Davidoff, B. Renault, "La reconnaissance des visages. II: Neuroscience cognitive", *éthologie et modélisation* 1994, vol. 39, no 4 (1 p.3-4), pp. 313-326
- [FIO96] N. George, J. Evans, N. Fiori, J. Davidoff, B. Renault Brain events related to normal and moderately scrambled faces. *Cognitive Brain Research*, 4(2), 65-76, 1996.
- [FIO97] N. George, B. Jemel, N. Fiori, B. Renault, "Face and shape repetition effects in humans: a spatio-temporal ERP study". *NeuroReport*, 8(6), 1417-1423, 1997
- [FOR03] D. A. Forsyth, J. Ponce, "Computer Vision – A Modern Approach" - Prentice-Hall, 2003
- [FRA96] C. Frankel, M. J. Swain, V. Athitsos, "WebSeer: An Image Search Engine for the World Wide Web", The University of Chicago Computer Science Department Technical Report 96-14 August 1996
- [GAB99] X. Gabaix, "Zipf's law for cities : An explanation", *Quarterly Journal of Economics* 114, 739–767, 1999
- [GOR04] I.E. Gordon, "theories of visual perception", Psychology Press, 2004, 256 pages
- [GUI02] E. Guichard, "L'internet : mesures des appropriations d'une technique intellectuelle", Thèse de doctorat de L'École des hautes études en sciences sociales (option sciences de l'information et de la communication), 2002
- [HAC67] A. P. Hackett, "70 Years of Best Sellers", 1895-1965, R.R. Bowker Company, New York, NY, 1967
- [HAF95] J. Hafner, H.S. Sawhney, W. Equitz, M. Flickner, and W. Niblack, "Efficient color histogram indexing for quadratic form distance functions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(7): 729-736, July 1995.
- [HAR85] D. Harwood, S. Muralidhara L.S. Davis, "Texture classification by local rank correlation", *CVGIP* 32, 1985 p 404-411
- [HAR88] C. Harris and M. Stephens, "A combined corner and edge detector", *Proceedings of the 4th Alvey Vision Conference*: pages 147—151, 1988
- [HIL96] T. Hill, "A statistical derivation of the significant-digit law", *Statistical Science* 10, 354-363, 1996
- [ICO78] L.D. Couprie, "Iconclass, a device for the iconographical analysis of art objects", *Museum* [published by Unesco] 25 (1978) 194-198
- [JAC92] A.E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations", *IEEE Transactions on Image Processing*, p 18-30, Janvier 1992
- [IPT07] IPTC white paper photo metadata, International Press Telecommunications Council, white paper 2007

- [JOL01] J. M. Jolion, J. Math, "Image and the benford's law", *Imaging Vision*, vol. 14, pp. 73-81, 2001.
- [LEL99] A. Lelu, M. Hallab, H. Rhissassi, F. Papy, S. Bouyahi, N. Bouhaï, H. He, C. Qi, I. Saleh, "Projet NeuroWeb : un moteur de recherche multilingue et cartographique, Méthodes" 5ème conférence Hypertextes et Hypermédias : Réalisations, Outils & (1999), H2PTM'99, Paris, 23 et 24 Septembre 1999
- [LI06] J. Li and J. Z. Wang, "Real-time Computerized Annotation of Pictures", *Proc. ACM Multimedia*: 911-920 (2006).
- [LUA91] E.T. Luand R.J. Hamilton, "Avalanches of the distribution of solar flares", *Astrophysical Journal* 380, p. 89–92 1991
- [MAN77] B. Mandelbrot., "Fractals: Form, Chance and DimenMon", W.H. Freeman and Co., San Francisco, 1977
- [MED05] G. Medioni, S. B. Sing Bing Kang, "Emerging Topics in Computer Vision". IMSC Press Multimedia Series, Prentice-Hall PTR, chapter 8, 2005
- [MEU03] M. Meunier, V. Nalwa, B. Bachevalier, "Reactions to familiar and novel objects in infant monkeys with neonatal temporal lesions", *Hippocampus*, 13 (4) 2003, p489-493
- [MIY00] S.Miyazima, Y.Lee, T.Nagamine, H.Miyajima, "Power law distribution of family names in Japanese societies", *PhysicaA* 278, 282–288, 2000
- [MOR04] M. Naaman, Y. Jiun Song, A. Paepcke, H. Garcia-Molina , "Automatically generating metadata for digital photographs with geographic coordinates", *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters New-York 2004*
- [MOR80] H. Moravec, "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover", Tech Report CMU-RI-TR-3 Carnegie-Mellon University, Robotics Institute, 1980
- [NEU94] G. Neukumand, B.A. Ivanov,, T. Gehrels, "Crater size distributions and impact probabilities on Earth from lunar, terrestrial planet, and asteroid cratering data", (ed.), *Hazards Due to Comets and Asteroids*, pp.359–416, University of Arizona Press, Tucson, AZ (1994)
- [NEW05] M. E. J. Newman, "Power laws, Pareto distributions and Zipf's law", *Contemporary Physics* 46, 323-351 (2005)
- [NIE07] R. Niels, L. Vuurpijl, L. R. B. Schomaker, "Automatic allograph matching in forensic writer identification", *International Journal of Pattern Recognition and Artificial Intelligence*, 21(1), p. 61-81, ISSN 0218-0014, 2007
- [PAQ04] S. Nicolas, T. Paquet, L. Heutte. "Enriching Historical Manuscripts: The Bovary Project", In *Proceedings of the 6th IAPR International Workshop on Document Analysis Systems*, Florence, (Italy), volume 3163 of *Lecture Notes in Computer Science*, pages 135–146, September 2004.
- [PER03] Per-Erik Forssén, Gösta Granlund, "Robust Multi-scale Extraction of Blob Features", *Lecture Notes in Computer Science*, Volume 2749/2003 p 769-780, ISBN978-3-540-40601-3

- [SAL71] G. Salton, "The SMART retrieval system: Experiments in Automatic Document Processing". Prentice Hall Inc., 1971
- [SAL00] P.Salembier, L.Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval", Image Processing, IEEE Transactions on, p561-576, April 2000
- [SCH97] B. Schiele, "Reconnaissance d'Objets utilisant des Histogrammes Multidimensionnels de Champs Réceptifs", Ph.D. Thesis, Institut National Polytechnique de Grenoble (1997)
- [SED98] M. J. Sedeyn, "introduction à l'examen objectif des écritures manuscrites", Editions Fovea, p. 110, 1998
- [SMA82] M. Smalland, J. D. Singer, "Resort to Arms: International And Civil Wars", 1816-1980. Sage Publications, Beverley Hills 1982 .
- [SMI96] J. R. Smith, S. F. Chang, "VisualSEEK : A Fully Automated Content-Based Image Query System", ACM Multimedia Conference, Boston, MA, USA, 1996.
- [SOR96] D. Sornette, L. Knopoff, Y. Y. Kagan, C. Vanneste, "Rank-ordering statistics of extreme events: application to the distribution of large earthquakes", Journal of Geophysical Research 101, 13883-13893, 1996
- [SRI01] S. N. Srihari, S.-H. Cha, H. Arora, S. Lee, "Individuality of handwriting : A validation study", 6th International Conference on Document Analysis and Recognition (ICDAR'01), Seattle, USA, September, p. 106-109, 2001.
- [STE92] F. Stein, G. Medioni, "Structural Indexing: Efficient 2D Object Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 12, pp. 1198-1204, Dec., 1992
- [STO98] R. Stoica, J. Zerubia, J.M. Francos, "Indexing and retrieval in multimedia libraries through parametric texture modeling using the 2D Wold decomposition", Research Report 3594, Inria, December 1998
- [STR95] A. Markus, B. Stricker and M. Orengo, "Similarity of Color Images", SPIE Proceedings Vol. 2420, 1995
- [SUR05] U.Surapong, M.Hammoud, C.Garrido, P.Franco, JM.Ogier, "Ancient graphic documents characterization", sixth IAPR International Workshop on Graphics Recognition August 25-26 2005 Hong-Kong
- [SWA91] M.J. Swain and D.H. Ballard, "Color indexing" International Journal of Computer Vision, 7(1): 11 32, 1991
- [TAN02] Y. Y. Tang, Yu Tao, E. C. M. Lam, "New method for feature extraction based on fractal behavior", Pattern Recognition, Vol. 35, Issue 5, P. 1071-1081, 2002
- [TEA80] M. Teague, "Image analysis via the general theory of moments", Journal Optical Society of America, 70:920-930, 1980.
- [TIM88] Timos K. Sellis, "Intelligent caching and indexing techniques for relational database systems", Information Systems Volume 13, Issue 2 (May 1988) Pages: 175 – 185

- [TOM02] K. Tombre, S. Tabbone, L. Plissier, B. Lamiroy, and P. Dosch, "Text/Graphics Separation Revisited", In D. Lopresti, J. Hu, and R. Kashi, editors, Proceedings of 5th IAPR International Workshop on Document Analysis Systems, Princeton (NJ, USA), volume 2423 of Lecture Notes in Computer Science, pages 200--211. Springer-Verlag, Aug. 2002
- [UTT05] S. Utama, J.M. Ogier, and P. Loonis, "Top-down segmentation of ancient graphical drop caps: Lettrines", In Workshop on Graphics Recognition (GREC), pages 87-96, 2005
- [VAL06] E. Valle, M. Cord, S. Philipp-Foliguet, "Content-Based Retrieval of Images for Cultural Institutions Using Local Descriptors," gmai, pp. 177-182, Geometric Modeling and Imaging--New Trends (GMAI'06), 2006
- [VIN03] C. Vinette, "L'information visuelle efficace pour la reconnaissance de visages dans l'espace-temps", Thèse de doctorat de l'Université de Montréal, 2003
- [VIN93] L. Vincent "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms", Image Processing, IEEE Transactions on 1993 Vol 2 Issue 2 p 176-201
- [VIN95] N. Vincent, H. Emptoz, "A classification of writings based on fractals", Fractal Review in the Natural and Applied Sciences, M.M. Novak Editor, London, Chapman and Hall, p. 320-331, 1995.
- [VIN98] N. Vincent, S. Barbezieux, "Compression of handwriting images: a way to define a writing style", Vision Interface, p.347-354, 1998
- [WEC97] M. Wechsler, P. Sheridan, P. Schauble, "Multi-Language Text Indexing for Internet Retrieval", Proc. of the RIAO'97 Computer-Assisted Information Searching on the Internet, 217-232 Paris
- [WEI98] S. Weibel, "Dublin Core: a simple content description model for electronic resources", NFAIS Newsletter 1998; 40(7): 1-3
- [ZIP49] G. K. Zipf, "Human Behavior and the Principle of Least- Effort", Addison-Wesley, Cambridge, MA, 1949