Tatouage de données géographiques et généralisation aux données devant préserver des contraintes

Cyril Bazin

UNIVERSITÉ de CAEN/BASSE-NORMANDIE

U.F.R. : Sciences

ÉCOLE DOCTORALE : SIMEM

THÈSE

présentée par

Cyril Bazin

et soutenue

le 20 janvier 2010

en vue de l'obtention du

DOCTORAT de l'UNIVERSITÉ de CAEN

spécialité : Informatique et applications

(Arrêté du 7 août 2006)

Tatouage de données géographiques et généralisation aux données devant préserver des contraintes

MEMBRES du JURY

Mauro Gaio	Professeur	Université de Pau	(rapporteur)
Daniel Augot	Directeur de Recherche INRIA	École polytechnique de Palaiseau	(rapporteur)
Mokrane Bouzeghoub	Professeur	Université de Versailles	
Caroline Fontaine	Chargée de Recherche CNRS	LabSTICC - CID et Télécom Bretagne - ITI	
Jacques Madelaine	Maître de conférences	Université de Caen	
Jean-Marie Le bars	Maître de conférences	Université de Caen	
Étienne Grandjean	Professeur	Université de Caen	(directeur)

Mis en page avec la classe thloria.

Remerciements

Je tiens à présenter ma reconnaissance à tous mes directeurs de thèses, tant officiels qu'officeux. Merci à Brigitte Vallée qui a trouvé le financement sans lequel rien n'aurait été possible. Merci à Étienne Grandjean pour avoir repris le flambeau de l'encadrement et m'avoir soutenu jusqu'au bout. Merci à Jacques Madelaine avec lequel j'ai pu avoir des conversations parfois houleuses mais toujours enrichissantes. Merci à Jean-Marie Le Bars pour sa pertinence, ses idées, ses remarques et surtout pour m'avoir supporté au jour le jour durant tant d'années.

Je témoigne toute ma reconnaissance aux membres de mon jury de thèse. Merci à Mokrane Bouzeghoub pour avoir présidé mon jury et s'être intéressé à mes travaux. Merci à Daniel Augot, Mauro Gaio et Caroline Fontaine dont les remarques et les questions ont grandement améliorées la présentation et le contenu de ce manuscrit. J'adresse un second remerciement à Caroline Fontaine pour son humilité face aux tracas administratifs.

Je salue à la fois la rigueur et la souplesse des membres du bureau des doctorants ainsi de de l'école doctorale qui ont permis à la soutenance de se tenir à la date prévue malgré toutes nos erreurs.

Á tous le personnel du GREYC qui m'a accompagné durant la thèse, je dis merci. Je parle bien sur des permanents du laboratoire qui m'ont traité comme un collègue et dont j'ai pu entrevoir le travail difficile durant mon année d'ATER. Je parle aussi de tout le personnel administratif du laboratoire qui fait tourner la machine de façon si efficace et qui a su m'aider chaque fois que j'en ai eu besoin.

Pour mes collègues et amis docteurs et doctorants, dont la liste est trop longue pour être citée ici de façon exhaustive, je dis merci. Merci pour leurs idées et remarques sur mes travaux. Merci surtout pour toutes les conversations quotidiennes, certes souvent sans intérêt, mais tellement amusante et remoralisante. Si vous avez été un moteur pour moi, j'espére que j'en serai un pour vous.

Pour l'équipe de Kalibee, votre amitié a été une force pour me faire progresser sur des terrains qui m'étaient inconnus. Votre courage est un exemple. Puisse notre aventure continuer encore longtemps.

Pour tous mes amis en France et de par le monde, je pense à vous et n'oublie pas de vous remercier. Je vous paierai une bière pour fêter ça.

À ma famille qui m'a soutenu de loin pendant ces années j'adresse mes remerciements. Même si mes travaux vous sont toujours resté vagues, je sais que vous avez toujours été derrière moi.

Pour l'infinie patience dont tu fais preuve chaque jour avec moi, Céline je te renouvelle tout mon amour.

On présente souvent la thèse comme un austère travail de recherche. Au delà de cette expérience professionnelle unique, la thèse est une aventure personnelle. J'ai trouvé pendant mon doctorat beaucoup plus que ce que j'y suis venu cherché. Qu'est ce qu'on s'est bien marré ! En espérant que l'aventure continue sous d'aussi bons hospices...

Table des matières

Table des figures

Liste des tableaux

Introduction

Contexte

Ces dernières années, l'essor des réseaux *peer-to-peer* et l'augmentation du débit des connections Internet ont largement facilité l'échange des documents numériques. Aujourd'hui, il est aisé de partager des fichiers et, lorsqu'une version pirate d'un document est disponible sur Internet, on peut facilement la trouver et la télécharger.

Pour se prémunir de la copie illicite, des garde-fous logiciels ont été développés, les *DRM*[1]. Celles-ci utilisent des lecteurs spécifiques qui limitent l'usage du document, par exemple en fixant le nombre de lectures autorisées. Le problème dû à cette limitation s'est rapidement fait sentir auprès des utilisateurs. Ceux-ci n'ont pas accepté d'acheter des documents dont les usages sont limités. Rappelons juste la lettre ouverte de Steve Jobs «Thoughts on Music» [2] qui condamne l'utilisation de ce genre de protection.

Le tatouage de documents numériques, sujet de cette thèse, est une alternative pour protéger la propriété intellectuelle des ayant droits sur les documents numériques sans en limiter l'usage. Il s'agit d'une technique de dissimulation d'information qui utilise deux dispositifs : le premier introduit une marque (un filigrane) dans un document tandis que le second détecte la présence de cette marque. Chacun de ces dispositifs utilise une clé. C'est la connaissance de la clé qui permet au propriétaire du document d'extraire la marque et de faire preuve de l'antériorité de ses droits. Retenons bien que la marque est mêlée au contenu du document, elle n'en change pas le format et il ne s'agit pas d'ajouter une métadonnée. L'un des problèmes du tatouage est d'introduire une marque suffisamment présente pour identifier l'auteur du document sans pour autant nuire à son usage ni à sa valeur.

Dans le cadre qui nous intéresse, on souhaite pouvoir retrouver la marque même lorsque le document a été transformé. Pour les images, la compression ou l'application de filtres sont des transformations usuelles possibles. On souhaite aussi que la marque soit invisible pour rendre sa détection et sa suppression plus difficile pour un utilisateur malveillant. Enfin, et il s'agit selon nous de l'aspect primordial, la marque ne doit pas dénaturer le document. Il est en effet essentiel de ne pas dévaloriser le document lorsqu'on y introduit la marque.

Les schémas de tatouage ont été initialement développés pour les données multimédia (images,

[1] Digital Right Management
[2] http ://www.apple.com/hotnews/thoughtsonmusic

musiques, etc.). Il s'agit alors de ne pas modifier la perception humaine du document. Un modèle de la perception humaine permet de mesurer la perte de qualité engendrée par l'application de la marque. Dans cette thèse, nous nous sommes intéressés à un problème différent : le tatouage de données contraintes. Nous appelerons *données contraintes*, les données pour lesquelles on souhaite préserver des contraintes formelles fixées. Pour les données géographiques, la contrainte sera de préserver la topologie du document.

Cette thèse présente deux sujets majeurs. Le premier est tatouage de documents géographiques vectoriels, nous présenterons le schéma que nous avons construit pour ce type de document. Le second sujet est le développement d'un schéma de tatouage générique destiné aux données contraintes. Nous présenterons le schéma générique ainsi que deux implémentations pour deux types de données différentes : les données géographiques vectorielles et les bases de données relationnelles.

Contributions

Cette thèse a débuté par une collaboration entre le GREYC, le Cédric, le Lamsade, le Cogit et l'IGN au sein du projet Tadorne de l'ACI sécurité & informatique 2004. Ce projet, dont le nom est une abréviation de «tatouage de contrainte» a mené au développement de plusieurs schémas pour tatouer différents type de données géographiques vectorielles. Le schéma développé dans la thèse de J. Lafaye [Lafaye, 2007] est conçu pour tatouer de la couche de bâti tandis que le notre s'intéresse au tatouage des réseaux routiers, que l'on retrouve dans les navigateurs GPS par exemple.

L'intérêt principal de notre schéma est de préserver des contraintes métriques et topologiques du document lors du tatouage. Le schéma est aveugle, ce qui signifie qu'il est possible de détecter la présence de la marque sans avoir besoin du document original (c'est-à-dire du document non-tatoué). Nous avons validé expérimentalement la robustesse du schéma face à des transformations courantes telles que le découpage de la carte, les transformations géométriques (rotation ou translation) ou le «retatouage» (seconde application de l'algorithme de tatouage avec une clé différente). Les expériences nous ont aussi permis de déterminer les tailles limites des documents pour lesquelles on arrive à détecter la marque. Les algorithmes pour tatouer et détecter la présence de marque sont rapides, leur complexité est quasi-linéaire en le nombre de sommets du document à tatouer.

L'un des aspects important du schéma est de travailler localement sur de petites parties du document, que nous avons nommé *sites*, à la fois pour marquer le document et pour choisir l'emplacement de la marque. On extrait un ensemble de sites d'un document, certains d'entre eux sont sélectionnés en fonction d'une clé puis modifiés. Notons bien que les sites sont modifiés seulement lorsque cela ne viole pas de contraintes sur le document et que cette vérification est faite localement. On estime qu'un document est tatoué lorsque, avec la clé, on arrive à retrouver assez de sites modifiés.

Il nous a semblé intéressant de reprendre cette notion de sites pour retrouver les bonnes

propriétés du schéma lorsqu'on souhaite tatouer de nouveaux types de données. Cet objectif nous a amené à créer un schéma de tatouage générique pour les données contraintes qui soit à la fois rapide, aveugle, robuste. La notion de généricité signifie que le schéma ne dépend pas directement du type de documents tatoués. Pour chaque nouveau type de documents à tatouer, il faudra alors décider de ce que représente un site et implémenter quelques fonctions spécifiquement pour le type de documents à tatouer. La construction d'un schéma de tatouage pour les bases de données illustre bien l'intérêt d'une telle généralisation. Ce schéma doit préserver le résultat d'une requête de somme sur les enregistrements de la base.

En plus de fournir un moule pour concevoir de nouveaux schémas, nous avons montré qu'il est possible de travailler directement sur le schéma générique. L'étude d'un schéma générique a des retombées directes pour toutes les implémentations du schéma générique. Pour bien illustrer ce point, nous avons étudié la résistance du schéma générique lorsqu'on modifie une partie des sites.

Nous avons aussi développé deux protocoles de détection générique. Le premier protocole pose la base d'un web-service générique de détection de la marque. Le second propose à un propriétaire potentiel de convaincre un juge qu'un document est tatoué sans divulguer la clé. Ces deux protocoles peuvent s'appliquer aux différentes implémentations du schéma.

Plan de la thèse

Les contributions de cette thèse seront présentées en trois parties dont l'ordre reflète notre démarche scientifique.

La première partie de ce mémoire présente le vocabulaire propre au tatouage et cadre de notre domaine d'étude. Nous verrons que nous nous sommes particulièrement intéressés aux schémas robustes, 0-bit, aveugles et destinés aux données contraintes. Cela signifie que la marque doit résister aux transformations, que l'on ne souhaite pas introduire un message dans le document, que l'on a uniquement besoin de la clé pour effectuer la détection et que nous nous intéressons aux documents dont les contraintes à préserver lors du tatouage sont exprimées formellement. Nous présentons trois exemples qui illustrent bien cette classe de schémas de tatouage. Nous définissons ensuite les données géographiques vectorielles et donnons un état de l'art des schémas de tatouage pour ces données.

Dans la seconde partie, nous présentons notre schéma de tatouage. Nous verrons que l'approche consiste à trouver une propriété aléatoire locale aux sites dont nous déterminerons la distribution expérimentalement. L'objectif est d'utiliser cet notion d'aléatoire locale pour introduire un biais statistique global (dans le document) sur cette distribution. Nous définirons aussi une fonction de codage afin d'attribuer un nombre (ou une couleur) à chaque site. Les codages des sites et la clé déterminent l'emplacement du biais.

Nous validons ce schéma expérimentalement en utilisant deux jeux de données provenant de l'IGN : les routes et les limites des communes du Calvados. Nous verrons notamment que le schéma résiste très bien au découpage. Une étude statistique des sites du document montrera que

dans un document géographique, il existe des classes de même codages relativement importantes. Cela permet à un attaquant de détecter la présence du biais sans avoir besoin de la clé et éventuellement de supprimer la marque. Nous proposerons et expérimenterons une variante du schéma qui résout ce problème.

La dernière partie est consacrée à la généralisation du schéma pour les données contraintes. Après avoir présenté notre schéma générique, nous donnons une étude de l'influence de la modification de l'ensemble des sites sur la détection de la marque. Nous terminerons par une présentation des deux protocoles de détection conçus pour notre schéma générique.

Première partie

État de l'art

Chapitre 1

Introduction au tatouage de données numériques

Sommaire

L'exemple le plus connu de tatouage se trouve sur les billets de banque. On y trouve un filigrane invisible que l'on distingue uniquement lorsqu'on regarde le billet à la lumière. Le filigrane est une marque dont l'absence caractérise les faux billets.

On retrouve cette idée de marquage pour les documents numériques. Il en existe de multiples applications. On peut par exemple l'utiliser pour vérifier l'intégrité d'un document, pour y introduire des méta-informations (sa date de création par exemple) ou pour en déterminer la paternité. Nous n'avons considéré que la dernière de ces applications dans cette thèse.

Aujourd'hui les échanges par Internet simplifient énormément le transfert des documents numériques, même pour ceux de grosse taille. Des personnes ayant peu de considération pour les droits d'auteurs peuvent très facilement publier ou revendre des documents dont ils ne sont pas les auteurs. Le tatouage de documents numériques apporte une solution à ce problème.

Nous voulons que l'auteur d'un document soit en mesure de prouver la paternité de la création d'un document. En marquant un document avant sa publication, l'auteur pourra vérifier la

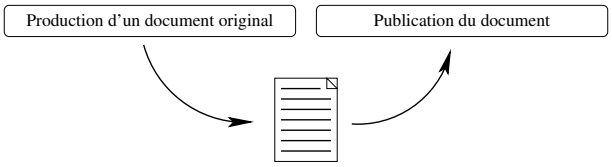


FIG. 1.1 – Exemple de cas d'utilisation de la publication d'un document.

présence de sa marque au sein de documents qu'il considère suspects. La détection de la marque pourra s'ajouter à un faisceau de preuves convergentes pour incriminer un suspect éventuel. On peut aussi envisager d'utiliser cette marque pour évaluer la dispersion d'un document sur Internet.

Ce chapitre va commencer par introduire le vocabulaire qui caractérise les différentes classes de schémas de tatouage numérique. Nous étudierons ensuite plus en détails une classe de schémas particulière : les schémas robustes, 0-bit, aveugles, destinés aux données contraintes. Nous donnerons enfin trois exemples de schémas qui illustrent bien le tatouage dans cette classe de schémas.

1.1 Présentation du tatouage de documents numériques

1.1.1 Principe du tatouage de documents numériques

La figure 1.1 illustre les étapes de la publication d'un document. Une fois un document créé, il est lancé dans le processus de publication et devient disponible pour le public. Une fois cette étape achevée, il n'existe pas de façon simple de prouver que le document appartient bien à son propriétaire légitime. Le tatouage de données est un moyen de remédier à ce problème. Il consiste à insérer une marque propre à l'auteur dans un document avant de le publier. Seul celui qui a inséré la marque, c'est-à-dire l'auteur du document, est capable de mettre en exergue cette marque et de faire preuve de sa propriété. La figure 1.2 illustre le processus de publication d'un document tatoué.

La marque insérée est mélée à la donnée originale. Il faut que le document soit suffisamment modifié pour que la marque insérée soit détectable et robuste, mais il faut aussi que la marque n'altère pas trop les données et en préserve les qualités. Cet équilibre rend le problème du tatouage particulièrement délicat. Notons bien qu'il faut concevoir une méthode de tatouage pour chaque type de document pour tenir compte de ses spécificités. Les méthodes de tatouages sont le plus souvent destinées aux documents multimédia tels que les images, les fichiers audio ou vidéo. Le tatouage de documents numériques s'étend aussi à d'autres types de documents. La liste est loin d'être exhaustive, citons tout de même le tatouage de textes [Atallah *et al.*, 2003], de modèles 3D [Wang *et al.*, 2007], de fichiers XML [Ng et Lau, 2005], de logiciels [Gupta et Pieprzyk, 2007] et de colorations de graphes [Qu et Potkonjak, 1998].

Le tatouage de documents numériques met en jeu deux algorithmes. Le premier consiste à insérer une marque, c'est l'*algorithme de marquage*. Le second, l'*algorithme de détection* permet de tester si une marque a été insérée dans un document. L'algorithme de détection doit permettre

FIG. 1.2 – Exemple de cas d'utilisation de publication d'un document tatoué.

de bien distinguer les documents tatoués de ceux qui ne le sont pas. L'ensemble de ces deux algorithmes est appelé *schéma de tatouage*.

Les processus de tatouage et de détection utilisent une *clé* pour identifier la personne qui a effectué le tatouage. Suivant le principe de Kerkhoffs [Kerkhoffs, 1883], c'est cette clé qui porte le secret, tous les algorithmes sont publics. Pour certains schémas de tatouage, seule cette clé est nécessaire pour détecter la présence de la marque. Ces schémas sont dits *aveugles*. À l'inverse, lorsqu'ils ont besoin, en plus de la clé, de se référer au document original, les schémas sont dits *non-aveugles*.

Beaucoup de schémas de tatouage utilisent cette clé pour insérer un message dans le document puis pour l'extraire. Dans la plupart des cas, la clé sert à chiffrer le message qui sera introduit. Le message peut être par exemple constitué du nom du propriétaire, d'une date et de méta-informations sur le document. Ces schémas sont dits *n-bits*. À l'inverse, les schémas que nous construirons sont dits *0-bits*, ils n'utilisent que la clé lors du marquage et de la détection. Ils n'insèrent pas de message dans le document.

Il est important de noter que la détection est susceptible d'échouer lorsque le document a été suffisamment modifié entre sa publication et la détection. Suivant le type de document tatoué, il est normal de considérer certaines transformations comme légitimes (compresser ou découper une image par exemple). Un schéma de tatouage est dit *robuste* lorsque la marque est préservée après application de ces transformations légitimes. Un schéma qui n'est pas robuste peut-être *fragile* ou *semi-fragile*. La marque disparaît alors quand le document est modifié. Ces schémas sont surtout utiles pour faire du contrôle d'intégrité. Comme le sujet de notre étude est la protection de la propriété intellectuelle, ces dernières classes de schémas ne seront pas abordés dans la thèse.

1.1.2 Tatouage de données multimédia ou bien de données contraintes

Il existe plusieurs domaines d'étude au sein de la communauté des scientifiques travaillant sur le tatouage de données numériques. Le domaine principal concerne le tatouage de données multimédia. La problèmatique est de concevoir un schéma qui ajoute une marque à un document multimédia tel qu'une image ou un fichier audio. Bien entendu, la marque insérée doit être robuste aux traitements classiques tels que la compression avec pertes. De plus, la marque ne doit pas modifier la qualité psychoperceptive du document. En d'autres termes, un être humain

regardant ou écoutant le document multimédia ne doit pas s'apercevoir d'une dégradation liée à la présence de la marque. Ces schémas utilisent différents modèles afin d'estimer la dégradation perceptuelle des documents.

À la différence du tatouage de données multimédia, le tatouage de données contraintes s'intéresse à préserver certaines qualités du document que l'on peut énoncer formellement. Cela permet par exemple de garantir que le résultat de certains algorithmes sera identique, que la donnée soit tatouée ou non, et aussi de vendre une donnée pour laquelle on peut fournir une garantie de qualité.

Ces deux domaines de recherche peuvent s'intersecter. Citons l'exemple du tatouage d'images médicales qui fait intervenir des données multimédia et des contraintes sur les traitements apportés aux images. Ainsi, les résultats d'éventuels algorithmes de traitement d'images utilisés pour détecter une tumeur ne doivent pas être modifiés par la marque insérée.

1.2 Définitions

Dans cette section, nous allons définir plusieurs classes de schémas de tatouage. Cela nous permettra de définir la classe de schéma qui nous intéresse : les schémas robustes, 0-bit, aveugles pour les données contraintes.

1.2.1 Schéma de tatouage

Un schéma de tatouage s'intéresse à une classe de document, notée $\mathcal{D}$. Il met en jeu deux fonctions $\mathscr{W}$ et $\mathscr{D}$ qui servent respectivement à insérer une marque et à détecter sa présence. Remarquons que les définitions de ces deux fonctions dépendent de la catégorie de schémas de tatouage à laquelle on s'intéresse. Par exemple, des schémas de tatouage aveugles ou non ne seront pas définis de la même façon car le schéma aveugle ne fait pas intervenir le document original lors de la détection. De même pour les schémas n-bits qui font intervenir un message quand les schémas 0-bit n'en ont pas besoin.

Dans tous les cas, il est nécessaire de faire intervenir une clé pour cacher l'information. Cette clé identifie la personne ayant effectué le tatouage sur le document, l'ayant droit du document. Tous les algorithmes présentés dans ce manuscrit utilisent une architecture basée sur des clés secrètes. Ils utilisent la même clé pour tatouer un document et vérifier si un document est tatoué.

1.2.2 Notion de préservation de qualité du document

Le tatouage de document implique une modification du document original. Cette modification peut engendrer une perte de qualité entre le document original et le document tatoué. Les schémas de tatouage de documents multimédias utilisent la notion de fidélité pour évaluer la dégradation psycho-perceptive due au tatouage [Cox *et al.*, 2001] [Painter et A., 2000] [Drelie Gelasca *et al.*, 2005]. L'évaluation de la fidélité fait intervenir un modèle perceptuel, et une

fonction δ pour mesurer la similarité entre le document original d et son tatoué w. On veut alors minimiser $\delta(d, w)$.

Le tatouage de données contraintes a pour objectif de préserver certaines qualités du document (que l'on peut énoncer formellement) entre le document original et son tatoué. Il faut pouvoir contrôler et borner la perte de qualité due à l'algorithme de tatouage. Nous n'avons donc pas besoin de mesure de similarité, nous devons juste être capable de savoir si les contraintes sont respectées entre le documents original et son tatoué. Pour définir cette notion de préservation de qualité, nous utiliserons une relation. Lorsque cette relation sera vérifiée, nous dirons que le document original et son tatoué sont de qualité équivalente.

1.2.3 Tatouage robuste ou fragile

Une fois un document publié, on peut facilement imaginer que des utilisateurs le modifient. Un utilisateur peut compresser, découper ou appliquer des filtres sur une image par exemple. Un schéma de tatouage est robuste à une transformation lorsqu'il est possible de retrouver la marque après avoir appliqué cette transformation sur le document. À l'opposé des algorithmes robustes, il existe des algorithmes de tatouages dits fragiles ou semi-fragiles. Pour ces algorithmes, la marque est effacée dès que l'altération des données est suffisamment importante. Ces schémas sont eux principalement utilisés pour contrôler l'intégrité d'un document. Comme ils ne sont pas utiles pour la préservation de la propriété intellectuelle, nous ne considérerons donc que les schémas robustes.

Comme le précise le livre [Cox *et al.*, 2001], il faut bien distinguer la notion de transformations légitimes de la notion d'attaques hostiles qui sont techniquement très différentes. Les transformations légitimes sont les traitements qu'un utilisateur risque d'effectuer dans le cadre de son travail quotidien. Contrairement aux transformations hostiles, elles n'ont pas pour objectif d'effacer la marque.

Avant de concevoir un schéma de tatouage, il faut lister les transformations légitimes car elles vont guider les choix de conception. Il est très difficile de rendre le schéma résistant aux transformations hostiles car il faut alors modéliser un attaquant (par ses buts et ses moyens) pour imaginer toutes les actions qu'il est capable effectuer.

Notons tout de même qu'il existe des outils pour valider la robustesse des schémas de tatouage d'images et audio. Citons le logiciel StirMark [Petitcolas *et al.*, 1998] [Petitcolas, 2000] [Raynal *et al.*, 2001] développé par F. Petitcolas qui évalue la robustesse d'un schéma grâce à un jeu de tests. Ce logiciel applique des transformations sur des documents tatoués puis vérifie que l'algorithme de détection retrouve toujours la marque.

1.2.4 Tatouage 0-bit ou n-bits

Nous avons vu que les schémas n-bits introduisent un message dans un document que l'algorithme de détection est capable d'extraire et de retourner. Cette approche a l'avantage de permettre l'ajout de méta-informations sur le document, sur son propriétaire ou sur l'heure de

tatouage par exemple. Le propriétaire peut faire preuve de ses droits lorsqu'il est capable de relire le message inséré. Cette approche présente un inconvénient majeur. Pour être relu correctement, il faut que l'ordre des bits du message résiste aux diverses transformations appliquées sur le document. Des problèmes de synchronisation peuvent aussi intervenir pour retrouver le début du message. Cependant, la répétition du message et l'utilisation de codes correcteurs peut améliorer la détection de la marque [Todorov, 2005] [Fontaine et Galand, 2008].

Le tatouage 0-bit n'introduit aucun message dans le document. Il le modifie de sorte que l'algorithme de détection retourne «vrai» quand la clé permet d'observer une propriété du document qu'il est très improbable d'observer sans elle. Nous avons choisi de travailler sur des schémas 0-bit pour nous affranchir des problèmes d'ordonnancement et de synchronisation.

1.2.5 Tatouage aveugle ou non

Nous avons vu qu'un schéma de tatouage est aveugle lorsqu'il n'a pas besoin du document original pour détecter la marque. Dans le cas d'un schéma non-aveugle, la marque est calculée par différence entre le document tatoué et l'original (non-tatoué). Cette classe de schémas offre un espace d'information beaucoup plus grand que les schémas aveugles. Cependant, la nécessité de devoir conserver le document original demeure un inconvénient majeur.

Les schémas aveugles permettent de tatouer un document et de vérifier la présence de la marque insérée sans avoir besoin d'effectuer de comparaison avec le document original. Bien qu'ils soient plus difficiles à concevoir, ces schémas présentent de multiples avantages. Tout d'abord, on évite de devoir divulguer à un tiers le document original au moment de la détection. De plus, on peut détruire le document original une fois le marquage effectué. On peut alors faire en sorte qu'il n'existe pas de copie du document original publié sans tatouage. Enfin, le tatouage aveugle permet de diminuer les coûts de stockage du document original, qui, s'il est conservé doit être stocké dans un endroit sécurisé.

1.2.6 Schémas de tatouage robustes, 0-bit, aveugles pour les données contraintes

Nous travaillerons sur les schémas destinés aux données contraintes qui font intervenir la préservation de qualité de document. Nous commençons par proposer une définition de cette notion. Nous noterons $\mathcal{K}$, l'ensemble des clés utilisables pour tatouer un document.

On suppose que l'on dispose d'une relation $Q_{\mathcal{D}}$ entre deux documents de $\mathcal{D}$. La qualité d'un document $d \in \mathcal{D}$ est dite préservée par un schéma de tatouage lorsque $Q_{\mathcal{D}}(d, w)$ est vérifiée entre un document $d \in \mathcal{D}$ et son tatoué $w \in \mathcal{D}$.

Définition 1.2.1 (Schéma pour les données contraintes) *Soit $Q_{\mathcal{D}}$ la relation de préservation de qualité. Un schéma de tatouage est un schéma de tataouge pour les données contraintes lorsqu'il garantit que pour tout document $d \in \mathcal{D}$, et son tatoué $w \in \mathcal{D}$, $Q_{\mathcal{D}}(d, w)$ est vérifiée.*

Nous avons vu que nous travaillerons sur des schémas 0-bit et aveugles.

Définition 1.2.2 (Schéma de tatouage 0-bit et aveugles) *Un schéma de tatouage est 0-bit et aveugle lorsqu'il est composé d'une fonction de marquage $\mathscr{W} : \mathcal{D} \times \mathcal{K} \to \mathcal{D}$ et d'une fonction de détection $\mathscr{D} : \mathcal{D} \times \mathcal{K} \to \{0,1\}$ telles que pour un document D choisi uniformément dans $\mathcal{D}$ et un clé $k \in \mathcal{K}$, on a :*

$$\begin{aligned} \Pr\Big(\mathscr{D}(D,k) = 1 \Big| \nexists d' \in \mathcal{D} \quad D = \mathscr{W}(d',k)\Big) &\leq \varepsilon^+ \qquad \textit{(faux-positifs)} \\ \Pr\Big(\mathscr{D}(D,k) = 0 \Big| \exists d' \in \mathcal{D} \quad D = \mathscr{W}(d',k)\Big) &\leq \varepsilon^- \qquad \textit{(faux-négatifs)} \end{aligned}$$

On dira alors que $(\mathscr{W}, \mathscr{D})$ est un schéma de taouage pour les seuils $(\varepsilon^+, \varepsilon^-)$.

Notons que dans la suite du document, nous ferons la confusion entre fonction ($\mathscr{W}$, $\mathscr{D}$, etc.) et algorithme qui la calcule.

Dans la définition précédente, les valeurs ε^+ et ε^- représentent respectivement un majorant de la probabilité de détecter un faux-positif et un faux-négatif. Un faux-positif signifie que l'algorithme de détection, qui calcule $\mathscr{D}$, retourne «vrai» pour un document non-tatoué. À l'inverse, on parle de faux-négatif lorsque le détecteur retourne «faux» pour un document tatoué. On veut bien entendu minimiser les probabilités d'avoir des faux-négatifs et des faux-positifs.

Nous définissons maintenant la résistance du schéma à une transformation. Notons que les transformations sur le document risquent d'augmenter le nombre de détections manquées, c'est-à-dire le nombre de faux-négatifs.

Définition 1.2.3 (Notion de robustesse) *Un algorithme de tatouage est dit ε^--robuste à une transformation $\mathscr{T} : \mathcal{D} \to \mathcal{D}$ si, étant donné une variable aléatoire D distribuée uniformément sur $\mathcal{D}$:*

$$\Pr\Big(\mathscr{D}(\mathscr{T}(D),k) = 0 \Big| \exists d' \in \mathcal{D} \quad D = \mathscr{W}(d',k)\Big) \leq \varepsilon^-$$

Conclusion

Nous venons d'introduire le vocabulaire et les notions que nous reprendrons dans le reste du manuscrit. Dans le chapitre suivant, nous présenterons plusieurs schémas qui illustrent toutes ces notions. Notons bien que tous les schémas que nous avons conçu et qui sont présentés dans le manuscrit sont tous des schémas robustes, aveugles, 0-bit, destinés aux données contraintes.

Chapitre 2

Tatouage de données contraintes

Sommaire

Dans leur livre «Digital Watermarking»[Cox *et al.*, 2001] ainsi que dans l'article [Cox et Miller, 2002], I. Cox *et al.* situent les premiers travaux de tatouage de documents au brevet déposé par E. Hembrooke [Hembrooke, 1961]. Il s'agit d'insérer sur un disque vinyle un signal intermittent dans les fréquences inaudibles, aux alentours de 1Hz. Ce signal correspond à un message codé en Morse ajouté par un dispositif mécanique au signal original. Un autre dispositif mécanique utilise un filtre pour lire ce message sur un haut-parleur. Il est intéressant de noter la présence des deux dispositifs de marquage et de détection ainsi que la volonté d'obtenir une marque perceptuellement indétectable.

Toujours selon I. Cox *et al.* , les premiers travaux concernant le tatouage de documents informatiques ont débuté en 1979 avec la contribution de Szepanski qui a décrit des méthodes d'insertion et de détection automatique d'un code d'identification dans un document. Plus tard, en 1988, Holt *et al.* produisirent un nouveau brevet pour l'insertion et la détection automatique d'un message dans un signal audio. Ce n'est que dans les années 90 que l'intérêt pour le tatouage de données numériques a vraiment commencé. Cet intérêt a surtout porté sur le tatouage de documents multimédia tels que les images, les documents audio ou les vidéos. Outre le fait qu'historiquement le tatouage se soit initialement intéressé aux données multimédia, l'intérêt pour ce type de données a été amplifié par les consortiums et organisations privées qui ont bien compris l'importance de protéger DVD, œuvres audio, etc.

Notons que la plupart des schémas s'intéressant aux documents multimédias traitent le document comme un signal. Cette approche permet de minimiser l'impact psychoperceptif de la marque sur le document mais ne garantit pas d'en préserver des qualités définies formellement. C'est pourquoi elles ne sont pas adaptées au tatouage de données contraintes.

Dans cette partie, nous commencerons par décrire la méthode du patchwork utilisée pour le tatouage d'image et décrite par Bender *et al.* dans [Bender *et al.*, 1996]. Nous verrons que cette méthode traite l'image comme un ensemble de pixels et non pas comme un signal. Le schéma est robuste, aveugle, 0-bit, et préserve la luminance globale d'une image. Nous introduirons ensuite le travail d'Agrawal *et al.* [Agrawal et Kiernan, 2002] qui est issu de la méthode du Patchwork appliquée au tatouage de base de données. Notons que ce schéma robuste, aveugle et 0-bit ne s'intéresse pas à préserver de qualités globales de la base de données. Pour finir, nous présenterons une extension du travail précédent proposée par Gross-Amblard [Gross-Amblard, 2003]. Ici, l'auteur tatoue une base de données relationnelle avec un schéma robuste, aveugle, qui préserve des requêtes de somme sur la base de données. Il est important de bien comprendre ces travaux car ils ont largement inspirés nos propres travaux sur le tatouage de données géographiques et sur la construction du schéma générique.

2.1 Technique du patchwork

Parmi les diverses techniques de tatouage, il nous semble important d'introduire la technique du patchwork. Elle a été présentée par W. Bender [Bender *et al.*, 1996] dans le cadre du tatouage d'images. Ce travail a ensuite été beaucoup repris et amélioré notamment dans le cadre du tatouage de données audio par D.G. Hong [Hong *et al.*, 2002] et H. Kang *et al.* [Kang *et al.*, 2007].

Cet algorithme se base sur une modification dans le domaine spatial. C'est-à-dire qu'elle modifie directement la luminance des groupes de pixels pour introduire un biais statistique dans l'image. L'idée de ce schéma est de sélectionner une séquence de n couples de groupes de pixels en fonction de leurs coordonnées et d'un générateur pseudo-aléatoire initialisé par une clé. Pour chaque couple de groupe de pixels, on augmente la luminance du premier groupe par une valeur δ et l'on diminue la luminance de l'autre par la même valeur. Cette modification a pour effet d'augmenter la différence (au sens de soustraction) de luminance entre le premier groupe et le second. Cela introduit un biais statistique dans l'image. Si le document n'est pas tatoué, la somme de ces différences sur un grand nombre de couples doit être proche de 0. Par contre, si le document est tatoué, l'algorithme de détection pourra retrouver les couples de groupes de pixels modifiés en utilisant la clé et il détectera un biais dans la luminance des pixels. En effet, si le document est tatoué la somme des différences de luminance doit être proche de $2\delta n$.

L'algorithme de détection a uniquement besoin de la clé pour vérifier si un document est tatoué. La détection est donc aveugle. Par ailleurs, l'algorithme peut être rendu plus robuste en choisissant les couples à la fois par une clé et par des heuristiques basées sur de la reconnaissance de forme.

Dans le cadre du tatouage d'image, cette approche dans le domaine spatial a vite trouvé ses limites, notamment du fait du manque de robustesse de la marque aux transformations telles que la compression. Cependant, ce schéma est intéressant car son principe est à l'origine des travaux de R. Agrawal [Agrawal et Kiernan, 2002] dans le cadre du tatouage des bases de

Algorithm 1: Algorithme d'insertion de marque dans une table de base de données

```
Data: k : la clé secrète
Data: t : la table à marquer
Data: γ : 1/γ représente la proportion d'enregistrements tatoués
Data: ν : nombre d'attributs numériques de la table pouvant être modifiés
Data: ξ : nombre de bits modifiables dans un attribut
begin
    foreach enregistrement e de la table t do
        p ← la clé primaire de e ;
        Initialisation d'un générateur pseudo-aléatoire G par (p, k) ;
        if suivant(G)%γ = 0 then
            i ← suivant(G)%ν ;
            j ← suivant(G)%ξ ;
            v ← suivant(G)%2 ;
            if v = 0 then
                Mise à 0 du j-ème bit du i-ème attribut de e ;
            else
                Mise à 1 du j-ème bit du i-ème attribut de e ;
end
```

données relationnelles.

2.2 Tatouage de base de données relationnelles

Les travaux d'Agrawal, Haas et Kiernan concernant le tatouage de bases de données relationnelles traitent de documents structurés et sont à l'origine du travail de D. Gross-Amblard *et al.* concernant le tatouage de bases de données relationnelles avec préservation de contraintes. Cette méthode a été présentée dans [Agrawal et Kiernan, 2002] et [Agrawal *et al.*, 2003]. Elle présente certaines similitudes avec la méthode du Patchwork vu précédemment.

Dans ce travail, les auteurs voient une table de base de données comme un ensemble d'enregistrements. Le marquage s'effectue en modifiant chaque enregistrement individuellement, indépendamment des autres. Le schéma ne préserve pas de qualité globale de la base de données, par contre, il garanti de ne modifier que les bits de poids faible d'un attribut numérique de chaque enregistrement. L'algorithme de tatouage parcourt chaque enregistrement d'une table. Pour chacun d'eux, on initialise un générateur pseudo-aléatoire à partir de sa clé primaire et d'une clé secrète. Ce générateur permet de déterminer d'une part si l'enregistrement sera modifié et d'autre part la façon de le modifier. L'algorithme de détection effectue la même sélection et vérifie la valeur des enregistrements.

L'algorithme de tatouage de la figure 1 effectue les étapes suivantes :

Algorithm 2: Algorithme de vérification de la présence d'une marque dans une table de base de données

Data: k : la clé secrète
Data: γ : $\frac{1}{\gamma}$ représentes la proportion d'enregistrements tatoués
Data: ν : nombre d'attributs de la table pouvant être modifiés
Data: ξ : nombre de bits modifiables dans un attribut
begin
  foreach *enregistrement e de la table t* **do**
    $p \leftarrow$ la clé primaire de e ;
    Initialisation d'un générateur pseudo-aléatoire G par (p, k) ;
    $nbInstancesTotal \leftarrow 0$;
    $nbInstancesOk \leftarrow 0$;
    if $suivant(G)\%\gamma = 0$ **then**
      $i \leftarrow suivant(G)\%\nu$;
      $j \leftarrow suivant(G)\%\xi$;
      $v \leftarrow suivant(G)\%2$;
      $w \leftarrow$ valeur du j-ème bit du i-ème attribut de e ;
      $nbInstancesTotal \leftarrow nbInstancesTotal + 1$;
      if $v = w$ **then** $nbInstancesOk = nbInstancesOk + 1$
  $\tau \leftarrow$ seuil$(nbInstancesOk, nbInstancesTotal)$;
  if $(nbInstancesOk < \tau)$ *or* $(nbInstancesOk > nbInstancesTotal - \tau)$ **then**
    return $True$; `// Document suspect`
  return $False$; `// Document non-suspect`
end

– sélection d'une instance de la donnée originale en fonction d'une clé ;
– décision : une instance sélectionnée sera-t-elle modifiée ?
– modification de l'instance sélectionnée si besoin .

L'algorithme de tatouage introduit un biais statistique dans certaines instances choisies par une clé. Sans celle-ci, il est impossible de savoir quelles instances ont été modifiées.

La méthode présentée par l'algorithme de détection 2 est très similaire :

– sélection d'une partie de la donnée originale en fonction d'une clé ;
– décision, pour savoir si l'instance sélectionnée a été modifiée ;
– comparaison entre la valeur réelle de l'instance et sa valeur attendue.

Un document est tatoué si le taux de correspondance entre les valeurs réelles et les valeurs attendues est proche de 100%. Si ce taux de correspondance est proche de 50%, le document n'a pas de biais statistique. On peut donc considérer qu'il n'est pas tatoué. La distinction entre ces deux cas extrêmes se fait en choisissant un seuil.

La détection ne nécessite pas de référence au document original. La valeur associée à un

nom	age	taille
Albert	20	160
Bobby	32	181
Cédric	48	163
Dimitry	33	174
Émile	35	175

FIG. 2.1 – Exemple de la table “personnes”

enregistrement est comparée à une valeur calculée en utilisant la clé secrète et la clé primaire de l'enregistrement. Par conséquent, la détection de la marque est bien aveugle.

Cette méthode est très bien adaptée aux bases de données et supporte les modifications classiques de la base de données telles que l'insertion ou la suppression de quelques enregistrements. En effet, l'idée de voir le tatouage comme un biais statistique sur un échantillon d'instances indépendantes permet de voir ce biais conservé même si quelques instances sont ajoutées ou retirées. Cependant, cette approche ne tient pas compte de contraintes globales de qualité. Par exemple, elle ne permet pas de préserver la valeur moyenne des enregistrements d'une colonne pour une table donnée.

2.3 Tatouage de base de données avec préservation de requêtes de somme

Plusieurs schémas de tatouage de base de données se sont inspirés du schéma précédent. Nous nous intéresserons essentiellement au travail de D. Gross-Amblard [Gross-Amblard, 2003] repris par J. Lafaye [Lafaye, 2007]. Ce schéma est un schéma n-bits (il permet d'ajouter un message dans la base). L'aspect le plus intéressant du schéma est de tatouer une base de données tout en préservant le résultat de requêtes de sommes choisies.

Notons que les auteurs ont développé un outil performant de tatouage de base de données disponible en ligne [Watermill, 2009] [Constantion *et al.*, 2005] qui permet de tatouer des bases de données de grandes tailles. Le projet n'est malheureusement plus maintenu depuis 2007.

À partir de la table de la figure 2.1, il est possible de définir une requête qui empêche de modifier la moyenne des tailles des personnes ayant entre 30 et 40 ans de plus de 5 cm. Cette requête a la forme suivante :

```
global 5 on (
  select sum(taille)
  from personnes
  where age between 30 and 40
);
```

De plus, des requêtes locales à chaque attribut définissent la perturbation maximale appli-

cable à l'attribut. Par exemple, pour limiter la modification de la taille d'une personne à plus de 2cm, on écrira la contrainte suivante :

```
local 2 on personnes.taille;
```

Dans le cas où plusieurs requêtes fixent une perturbation maximale applicable à un attribut, seule la plus petite perturbation autorisée est retenue. Par exemple, à partir des deux requêtes suivantes, on déduit que la valeur maximale des perturbations concernant la taille de Cédric est de 2cm. Par contre, la perturbation maximale autorisée pour Albert est limitée à 1cm car son âge est inférieur à 30 ans.

```
local 1 on (
  select taille
  from personnes
  where age < 30
);
local 2 on personnes.taille;
```

Pour préserver une contrainte globale de somme sur une table, l'algorithme de tatouage reprend le schéma présenté précédemment. La différence réside dans le fait de prendre les lignes de la tables par paires au lieu de les prendre individuellement.

Dans un premier temps, tous les enregistrements participant aux même contraintes de somme globale sont regroupées. Ensuite, chaque ensemble d'enregistrements qui participe aux mêmes relations est trié en fonction d'un hachage d'une clé secrète et de la clé primaire de la table. On obtient un tri déterministe reproductible uniquement par le possesseur de la clé secrète. Enfin, les enregistrements triés sont regroupés deux par deux pour former des paires. Ces paires sont tatouées de façon à ce que la modification d'un bit dans un élément de la paire soit compensé par la modification du même bit dans l'autre élément. Bien entendu, pour qu'une telle compensation soit possible, il faut traiter uniquement les paires dont les deux bits à modifier sont différents. Ainsi, la somme des deux éléments d'une paire est identique avant et après tatouage et la somme globale est préservée.

L'algorithme 3 illustre la construction des paires à partir d'une table de base de donnée, des contraintes globales et d'une clé. Il est important de noter que les contraintes globales interviennent uniquement pour la création des paires d'enregistrements.

L'algorithme 4 qui présente la méthode de tatouage considère uniquement la notion de paires et n'utilise que des contraintes locales pour modifier les données. Comme cet algorithme prend un message en paramètre, le schéma est bien n-bits. Cet algorithme est très proche de celui proposé par Agrawal. Connaissant la méthode de détection utilisée par Agrawal et la méthode de tatouage utilisée par Gross-Amblard, l'algorithme de détection est évident : il suffit d'utiliser la clé pour reproduire les paires puis pour relire le message.

Les auteurs donnent une étude très détaillée du schéma qui valide sa robustesse. Nous constatons juste que, comme le schéma est un schéma n-bits, des problèmes de synchronisation peuvent

Algorithm 3: Algorithme d'appariement d'enregistrements d'une table : CalculePaires

```
Data: k : la clé secrète
Data: t : la table à tatouer
Data: G : l'ensemble des contraintes globales
begin
    Créer les groupes d'enregistrements g_1, ..., g_n de t participant aux mêmes contraintes
    globales G ;
    paires ← ∅ ;
    foreach Groupe g_i do
        Trier g_i en fonction d'un hachage de k et de la clé primaire de chaque
        enregistrement.  while taille(g_i) ≥ 1 do
            Extraire le premier élément de g_i dans e_1 ;
            Extraire le premier élément de g_i dans e_2 ;
            Ajouter (e_1, e_2) à paires ;
    return paires
end
```

subvenir. On peut en effet perdre le couplage des enregistrements. C'est par exemple le cas lorsque l'enregistrement situé au début de la liste triée lors du tatouage est supprimé. Si on ne peut reconstituer le couplage, la détection échoue.

Retenons qu'il s'agit d'un schéma aveugle, robuste et n-bit qui préserve une qualité globale au niveau de la base par compensation de modifications locales. De plus le tatouage et la détection sont relativement rapides, l'opération qui coûte le plus en terme de complexité est le tri des enregistrements au début de l'algorithme.

Conclusion

Les schémas que nous avons présenté dans ce chapitre appartiennent tous à la classe des schémas de tatouage de données contraintes robustes, 0-bit et aveugles.

Remarquons que tous ces schémas utilisent plus ou moins une notion de localité. Dans le cas du tatouage d'image, on agit sur des couples de groupes de pixels et pour les bases de données, le travail s'effectue à l'échelle d'un ou deux enregistrements.

Nous retiendrons d'une part que ce sont de légères modifications locales qui forment la marque à l'échelle du document, et, d'autre part que c'est la clé qui détermine où ces modifications interviennent.

Toutes ces méthodes ont inspiré l'élaboration du schéma destiné aux données géographiques présenté dans la partie II et la généralisation aux données contraintes présentée dans la partie III.

Algorithm 4: Algorithme d'insertion de marque dans une table de base de données

Data: k : la clé secrète
Data: t : la table à tatouer
Data: G : l'ensemble des contraintes globales
Data: L : l'ensemble des contraintes locales
Data: m : message à insérer
Data: γ : $\frac{1}{\gamma}$ représente la proportion d'enregistrements tatoués
Data: ν : nombre d'attributs de la table pouvant être modifiés
Data: ξ : nombre de bits modifiables dans un attribut
begin
  $paires \leftarrow$ CalculePaires(k, t, G) ;
  foreach *paire d'enregistrements* (e_1, e_2) *de paires* **do**
    $p \leftarrow$ la clé primaire de e_1 ;
    Initialisation d'un générateur pseudo-aléatoire P par (p, k) ;
    if $suivant(P)\%\gamma = 0$ **then**
      $j \leftarrow$ une puissance modifiable de e_1 et e_2 respectant L ;
      $k \leftarrow suivant(P)\%taille(m)$;
      if *j-ème bit modifiable de* e_1 *et* e_2 *sont différents* **then**
        $v \leftarrow suivant(P) \oplus m[k]$;
        Mise à v du j-ème bit modifiable de e_1 ;
        Mise à $\neg v$ du j-ème bit modifiable de e_2 ;
end

Chapitre 3

Tatouage de documents géographiques

Sommaire

Dans ce chapitre, nous abordons la problèmatique du tatouage de documents géographiques vectoriels. Nous présentons les données géographiques considérées ainsi qu'une sélection de travaux de ce domaine.

3.1 Les données géographiques

On appelle donnée géoréférencée un ensemble de données géométriques et de données descriptives utilisées dans une application en géomatique. Les données géométriques renseignent sur la position et la forme d'une entité. Les données descriptives sont relatives aux attributs des entités. Si l'on traite une base de données routières, chaque tronçon de route est une entité. La géométrie de cette entité représente le tracé de la route. Les attributs peuvent être, par exemple, le nom de la route, son nombre de voies et sa vitesse maximale autorisée.

De nos jours, ce type d'information est de plus en plus utilisé. Les outils tels que les navigateurs GPS basent leurs calculs sur les cartes vectorielles. Les sites internet tels que Mappy, Googlemap et le site de l'IGN Géoportail fondent leur valeur ajoutée sur la pertinence de l'information géographique sur laquelle ils basent leurs calculs. Les SIG (Systèmes d'Information Géographique) facilitent énormément l'accès et les traitements sur les données géographiques. Ces exemples sont très loin d'être exhaustifs et l'on voit de plus en plus de sites internet et de

programmes qui utilisent la donnée géographique, afin de tirer parti des GPS sur les portables par exemple.

Ces données sont rassemblées par des organismes spécialisés tels que l'IGN. Dans ce cas, des équipes de professionnels recueillent ces données à partir de photos aériennes et de relevés terrain. Ce processus demande du temps et énormément de ressources humaines. C'est ce qui donne de la valeur à la base de données construite et qui explique que cette donnée est chère. Nous pouvons noter que des initiatives comme *OpenStreetMap* [OpenStreetMap, 2009] visent à construire une base de données routières libre de façon collaborative. Chaque personne peut ajouter de l'information obtenue par son GPS personnel dans une base de données commune au niveau mondial. Cette approche très intéressante est pourtant peu sûre. En effet, la donnée est incomplète, pas forcément à jour et des erreurs de relevés peuvent facilement se produire.

Pour ces raisons, il est souvent préférable d'utiliser des données géographiques provenant d'organismes tels que l'IGN qui garantissent une certaine qualité. Cependant, le prix de ce type de données peut inciter certaines personnes à les utiliser sans rétribuer les organismes qui les ont collectées. Par conséquent il est pertinent de poser le problème de la protection de la propriété intellectuelle sur les données géographiques. Nous pensons que le tatouage de données est une façon efficace de lutter contre ce problème. En effet, il devient plus facile pour un propriétaire de faire preuve de ses droits devant un tribunal en utilisant le tatouage. Nous pensons que cela peut contribuer à décourager la revente et la dissémination illicite des données géographiques.

3.2 État de l'art du tatouage de données géographiques

Les données géographiques peuvent prendre plusieurs formes. Il peut s'agir, entre autres, d'images bitmap provenant de satellites ou de données vectorielles provenant de relevés sur le terrain. Cet état de l'art ne présente que des méthodes de tatouages appliquées aux données vectorielles. Ce type de données, que l'on retrouve par exemple dans les navigateurs GPS, est spécifié par le consortium OGC [Open Geospatial Consortium, 2002] qui regroupe un grand nombre des acteurs majeurs du domaine.

Un document géographique vectoriel est constitué d'un ensemble d'objets géographiques où chacun peut représenter par exemple une route ou une ville. Chaque objet est constitué d'un ensemble de propriétés que l'on nomme aussi données descriptives et d'un ensemble de géométries. Les données descriptives peuvent être, pour une ville, sa population, le nombre d'actifs, le nombre d'usines dans la ville, etc. Les géométries représentent l'emprise de l'objet sur la terre et sont composées de trois types géométriques primitifs : le point, la ligne et le polygone. Un document est donné avec un système de projection et une précision qui représente l'écart maximal entre un point dans le monde réel et son image dans la base de données.

Notons que beaucoup de travaux de tatouage de la littérature ne prennent pas en compte la notion d'objet géographique. Dans ces schémas, les documents géographiques vectoriels sont vus en tant que cartes vectorielles. Cela supprime toute notion d'objet géographique et de géométries et les auteurs travaillent sur un graphe constitué d'un ensemble de sommets et d'arêtes entre

ces sommets. Cette approche se justifie pour deux raisons. Tout d'abord, la partie géométrique des documents géographique offre un espace suffisant pour poser une marque. Par ailleurs, il est facile de passer d'un document géographique vectoriel à un graphe. L'opération inverse, qui consiste à retrouver exactement le document géographique original après l'avoir transformé en graphe est par contre très difficile, voire impossible à cause de la perte d'information. Or, il est intéressant de pouvoir tester si une carte vectorielle, trouvée sur internet par exemple, est issue d'un document géographique préalablement tatoué. Les schémas qui ne tiennent pas compte de la notion d'objet géographique permettent donc vérifier à la fois si un ensemble d'objets géographiques et si une carte vectorielle issue de cet ensemble sont tatoués.

Pour présenter cet état de l'art du tatouage de données géographiques nous avons choisi de respecter la catégorisation de X.M. Niu *et al.* dans leur état de l'art du tatouage de cartes vectorielles [Niu *et al.*, 2006]. Dans un premier temps, nous présentons les méthodes basées sur des transformées. Nous présentons ensuite les méthodes issues du tatouage de modèles 3D et les méthodes qui travaillent dans le domaine spatial. Enfin, nous dresserons le bilan de ces schémas sous différents aspects.

3.2.1 Schémas basés sur des transformées

Les transformées sont utilisées dans le domaine de la compression et souvent reprises par les schémas de tatouage de données multimédia. Elles permettent d'extraire des coefficients caractéristiques à partir d'une donnée. Un schéma de tatouage basé sur une transformée opère, non pas directement au niveau de la position des sommets du document, mais sur des coefficients caractéristiques. L'algorithme de tatouage applique de légères modifications aux coefficients. L'algorithme de détection extrait à nouveau les coefficients pour vérifier s'ils ont été modifiés. Les schémas abordés dans cette section utilisent les transformées discrètes de Fourier (DFT), en ondelette (DWT) ou en cosinus (DCT).

Schémas de tatouage utilisant les DFT

N. Nikolaidis *et al.* [Nikolaidis *et al.*, 2000] [Solachidis *et al.*, 2000] proposent de travailler au niveau des géométries présentes dans le document. La méthode s'applique pour le tatouage de polygones. Le schéma de tatouage est aveugle et robuste.

Cette méthode est basée sur une propriété intéressante de la transformée de Fourier discrète (DFT). Celle-ci calcule un ensemble de nombres complexes à partir des coordonnées des sommets composant le polygone. Or, la norme de ces nombres complexes reste invariante si le polygone est translaté, tourné, si l'on change le premier sommet du polygone ou si l'on applique n'importe quelle combinaison de ces transformations. Par ailleurs, appliquer un effet miroir sur l'axe des x ou des y inverse juste l'ordre dans lequel apparaissent les nombres complexes.

La marque est introduite en modifiant la norme de certains des complexes, ainsi elle peut résister à toutes les transformations citées précédemment. En pratique, une suite de valeurs pseudo-aléatoires générée à partir d'une clé secrète sert à modifier la norme des complexes choisis.

En utilisant la tranformée de Fourier inverse, on obtient un ensemble de nouvelles coordonnées pour les points composant le polygone. La probabilité de présence de la marque peut ensuite être estimée de façon aveugle. Il suffit de calculer la corrélation entre les complexes choisis, que l'on extrait du document à vérifier, et les valeurs issues du générateur pseudo-aléatoire, que l'on recalcule à partir de la clé secrète. En utilisant un seuil, il est possible de décider si un document a été tatoué ou non par une clé donnée.

La détection peut encore être améliorée par l'utilisation du théorème de Bayes. Un nouvel algorithme de détection est présentée par V.R. Doncel et certains des auteurs de la méthode originale [Doncel *et al.*, 2005] [Doncel *et al.*, 2007]. Une astuce rend l'algorithme robuste à l'effet miroir.

Par construction, le schéma est robuste à toute combinaison de transformations basées sur la translation, la rotation ou le changement de premier sommet du polygone. Cependant, le schéma n'est pas robuste à des transformations telles que la suppression de sommets ou l'interpolation. De plus, le calcul de la transformée de Fourier, effectué lors du tatouage et de la détection, devient relativement coûteux pour des polygones ayant beaucoup de sommets. Enfin, l'algorithme ne donne aucune garantie quand au déplacement des sommets lors de la phase de tatouage. On ne peut donc pas borner le déplacement d'un sommet, ce qui est très génant si l'on veut donner une garantie sur la précision du document. De plus, on ne peut pas contrôler la distortion du polygone.

Les auteurs ont proposé une variante du schéma [Nikolaidis *et al.*, 2000] où il ne s'agit plus de tatouer un polygone, mais un ensemble de polygones. Ce schéma utilise l'algorithme précédent pour chaque polygone du document géographique. On obtient ainsi autant de coefficients de corrélation que de polygones dans le document de départ. Pour décider si l'ensemble du document est tatoué, les auteurs proposent différentes méthodes pour fusionner ces coefficients de corrélation. Ce schéma offre une meilleure détection, mais il présente toujours les même points faibles que la version initiale. Une dernière méthode [Kitamura *et al.*, 2001] utilise la transformée de Fourier, cette fois ci pour introduire un message au sein du document. Cependant, ce schéma n'est pas aveugle et il n'est robuste ni à la simplification, ni au découpage de la carte.

Schémas utilisant la DWT

Y. Li *et al.* ont proposé dans [Li et Xu, 2003] un schéma aveugle et robuste qui permet d'inclure un message au sein d'un document géographique. Leur schéma est basé sur la transformée en ondelelette discrète (DWT).

Cette transformée utilise l'ensemble des coordonnées des sommets du document afin de calculer un certain nombre de coefficients représentatifs ordonnés du plus représentatif au moins représentatif. Ces coefficients sont décomposés en quatre groupes dont chacun représente un niveau de détails. L'algorithme néglige les coefficients du groupe de coefficients les moins représentatifs car ils sont très sensibles aux modifications du document. Les coefficients du groupe de coefficients le plus représentatif n'est pas modifié non plus car cela aurait trop d'impact sur le

document. L'algorithme ne change que les coefficients des deux groupes intermédiaires.

Un message est inséré dans un groupe de coefficients en modifiant les coefficients d'ordre impair et en utilisant les coefficients d'ordre pair en tant que référence. On insère ainsi un bit de donnée dans chaque coefficient d'ordre impair. La valeur du bit inséré vaut 1 si et seulement si la parité de la différence entre l'amplitude du coefficient et la moyenne des amplitudes de ses deux voisins est impaire. Ainsi, le bit peut être relu lors de phase de détection sans avoir besoin du document original. L'algorithme utilise l'opération inverse de la DWT afin de retrouver, à partir des 4 groupes de coefficients, un ensemble de coordonnées tatouées.

Le schéma est aveugle, robuste aux transformations géométriques. Cependant, il ne permet pas de borner le déplacement d'un sommet. Enfin, le schéma ne résiste ni à la simplification ni à l'interpolation de sommets.

Schémas basés sur la DCT

Le schéma de tatouage proposé par M. Voigt *et al.* dans [Voigt *et al.*, 2004] et [Voigt *et al.*, 2005] utilise la transformée en cosinus discrète (DCT) pour tatouer des documents géographiques. Le schéma propose un tatouage aveugle, robuste. Il est aussi réversible : en connaissant la clé, on peut supprimer la marque et retrouver exactement le document original. On trouve des exemples de schémas de tatouage réversibles pour les données médicales, par exemple dans [Coatrieux *et al.*, 2008]. La transformée en cosinus discrète permet de calculer un ensemble de coefficients qui représente la corrélation entre des données. En effet, les auteurs font l'hypothèse que dans un document géographique, les positions d'un groupe de sommets proches sont hautement corrélées.

Ici, les auteurs considèrent le document géographique comme un ensemble de sommets et ne tiennent pas compte du tout de la notion d'objets géographiques. Les sommets sont regroupés par groupes de huit. Dans chaque groupe on stocke un bit de la marque. Le bit est inséré dans un groupe en augmentant les coefficients les moins significatifs du groupe. Cette augmentation des coefficients peut ensuite être retrouvée lors de la détection.

Notons que les auteurs ont aussi proposé une amélioration de cet algorithme pour borner le déplacement maximum d'un sommet sur chaque axe.

Cet algorithme est robuste, aveugle, réversible et permet d'inclure un message dans le document. De plus, il permet de borner le déplacement des sommets. Enfin, le calcul de la DCT n'est effectué que sur de petits groupes de sommets de tailles bornées . Cela permet un tatouage et une détection en un temps raisonnable, linéaire en la taille du document. Cependant le tatouage ne résiste pas au découpage de la carte, ni à l'interpolation de sommets.

3.2.2 Schémas basés sur le tatouage d'objets 3D

R. Ohbuchi *et al.* ont commencé par proposer des schémas de tatouage d'objets 3D [Ohbuchi *et al.*, 2002a] [Ohbuchi *et al.*, 2001] qu'ils ont ensuite adaptés au tatouage de documents

géographiques [Ohbuchi *et al.*, 2003]. Le schéma est robuste, non-aveugle et permet d'introduire un message au sein du document en utilisant une analyse spectrale.

Le schéma ne travaille pas sur le document lui même, mais sur la triangulation de Delaunay des sommets du document. Cela se justifie pour plusieurs raisons. Tout d'abord, la triangulation de Delaunay permet aux auteurs de se rapprocher du tatouage d'objets 3D. Ainsi, pour les deux types de données il s'agit de tatouer un maillage de triangles. De plus, elle fournit une notion de voisinage des sommets qui est corrélée à la topologie du document. Enfin, les auteurs vont préserver cette triangulation lors du tatouage. Comme elle est très représentative de la topologie du document, ils vont ainsi limiter la dégradation du document.

Le schéma utilise un *quadtree* pour découper la triangulation du document géographique en un ensemble de rectangles contenant un nombre de sommets borné. Chacun de ces rectangles, que l'on appelle *patch*, est ensuite traité indépendamment des autres. Chaque *patch* est vu comme un graphe connexe. On peut extraire les valeurs propres et les vecteurs propres de ce graphe via une analyse spectrale basée sur la matrice Laplacienne extraite du graphe. On obtient ainsi autant de couples de coefficients spectraux (ordonnées) que de sommets dans le *patch* original. L'algorithme de tatouage va modifier ces coefficients en y introduisant les bits du message chiffré qui identifie l'auteur du document. Pour coder un bit à 1, on ajoute une valeur α au coefficient. Pour coder un bit à 0, on retire une valeur α au coefficient. La valeur α représente l'intensité de la marque. Plus elle est élevée, plus la marque sera résistante, mais elle sera aussi plus visible et dégradera davantage le document. La dernière étape de l'algorithme consiste à reconstruire un « patch » tatoué en partant des valeurs propres et des vecteurs propres modifiés.

Lors de la détection de la marque, le document à vérifier est superposé au document original pour reproduire les « patches ». Pour chaque « patch », les coefficients spectraux sont calculés et comparés aux coefficients du « patch » correspondant dans le document original. Le signe de la différence entre un coefficient provenant du document original et celui correspondant dans le document à vérifier produit un bit. L'ensemble des bits obtenus donne un message. Si le message lu est celui attendu, le propriétaire légitime du document pourra faire preuve de ses droits.

Le schéma demande le calcul d'une triangulation de Delaunay et une analyse spectrale pour chaque « patch ». Le calcul de la triangulation peut être effectuée en temps quasi-linéaire. Rappelons aussi que les patches sont de taille bornée. Lors de la phase de détection, il faut ajouter le calcul d'un isomorphisme de graphes pour synchroniser le document à vérifier et le document original. Le calcul d'isomorphisme du graphe est rapide car le graphe est planaire. Le schéma, sans être particulièrement efficace, reste relativement raisonnable.

Par construction, le schéma est insensible à la translation ou la rotation du document. Il est robuste à l'insertion de sommets, l'ajout de bruit, au changement de l'ordre des objets ou au changement d'échelle. L'un des principaux intérêts du schéma est d'être résistant au découpage de la carte. Par contre, ce schéma est sensible à une seconde application de l'algorithme de tatouage avec une clé différente. En effet, si l'on tatoue par dessus un document tatoué, la seconde marque peut complétement effacer la première. Enfin, le schéma n'est pas aveugle car le document original est nécessaire lors de la phase de détection. Certains problèmes demeurent

pour que ce schéma puisse être utilisé en pratique. L'aspect non-aveugle du schéma soulève notamment des questions que les auteurs n'abordent pas. En effet, pour prouver que l'on est l'auteur d'un document, on le compare à un document de référence. Ainsi, l'attaquant peut forger un document de référence. Celui-ci serait construit de sorte qu'en le comparant au document tatoué, on retrouve le message de l'attaquant. On retrouve donc à la fois le message de l'attaquant et celui du propriétaire légitime dans le document tatoué. Les auteurs ne mentionnent pas ce type d'attaque.

3.2.3 Les méthodes basées sur des modifications géométriques

Les méthodes de tatouage basées sur des modifications géométriques introduisent une marque dans un document, soit en modifiant directement les positions de certains sommets choisis, soit en introduisant de nouveaux sommets dans le document. La plupart du temps, un ensemble de sommets est sélectionné. La modification de la position des sommets dans cet ensemble permet d'introduire une marque. Les schémas qui utilisent l'ajout de sommets travaillent sur les arêtes du document. La marque est introduite en ajoutant de nouveaux sommets le long des arêtes. Dans cette section nous présentons un certain nombre de schémas de tatouage qui agissent selon ces deux modèles.

Les méthodes basées sur le déplacement de sommets

H. Kang [Kang *et al.*, 2001] découpent le document en zones de surfaces égales. Dans chaque zone, on utilise un masque pour sélectionner un certain nombre de sommets. Chaque zone est ensuite découpée en deux en suivant la droite qui passe par le point le plus au sud-ouest et celui le plus au nord-est parmi ceux qui sont contenus dans le masque. On obtient ainsi deux ensembles de sommets (les sommets du nord-ouest et ceux du sud-est) qui permettent de coder un bit d'un message chiffré. Pour coder un bit à 0, on déplace les sommets appartenant au premier ensemble vers leur position symétrique par rapport à la droite. Pour coder un bit à 1, on effectue la même opération mais avec les sommets appartenant au second ensemble. Lors de la phase de décodage, on retrouve les deux ensembles de sommets. On lit un bit à 0 ou à 1 suivant l'ensemble majoritaire. On obtient ainsi un message que l'on peut comparer avec le message chiffré original. Le schéma est aveugle, et permet d'introduire un message dans le document. Il est robuste à l'ajout de bruit et à la suppression de sommets. Les auteurs utilisent un test de PSNR (Peak Signal To Noise Ratio) pour vérifier que le bruit introduit par le tatouage reste faible. Mais cette mesure, adaptée au traitement d'images, n'a que peu d'intérêt dans le cadre du tatouage de données géographiques. En effet, il est plus pertinent, dans le cas des données géographiques de contrôler la dégradation que l'algorithme fait subir au document pendant le processus de tatouage, plutôt que de la mesurer après coup. Cet aspect n'est pas du tout pris en compte par les auteurs.

Dans l'article [Ohbuchi *et al.*, 2002b], Ohbuchi *et al.* présentent un schéma de tatouage basé sur un déplacement direct des sommets qui n'est pas aveugle. Les auteurs proposent trois

façons distinctes de découper le document. La première propose un découpage en rectangles uniformes. La seconde utilise les quadtrees pour obtenir un certain nombre de rectangles. La dernière méthode est une variante qui consiste à effectuer un découpage en quadtrees puis à regrouper les rectangles qui ne contiendraient pas assez de sommets. Ces méthodes permettent d'insérer un nombre variable de bits dans le document, la dernière insérant le plus de bits. Quelle que soit la méthode choisie, chaque rectangle est ensuite traité de la même façon. On veut insérer un message chiffré, un maximum de fois dans un rectangle. On insère un bit pour chaque coordonnée de chaque sommet. Les sommets sont traités en suivant un ordre ; pour insérer un bit on déplace le sommet d'une certaine amplitude paramétrée par l'utilisateur. C'est cette amplitude qui borne le déplacement des sommets du document. Le schéma nécessite le document original pour effectuer la détection, il n'est donc pas aveugle. Après une étape de synchronisation entre le document original et le document à vérifier, on peut redécouper le document en rectangles. Pour décoder un bit, on compare la position du sommet avec la position du sommet correspondant dans le document original. On retrouve ainsi un message que l'on compare au message inséré. Le schéma est résistant à la translation, au changement d'échelle, à la modification de l'ordre des objets, à la suppression et à l'ajout de sommets. Il est relativement résistant à l'ajout d'un faible bruit et au découpage du document. Dans l'article qui présente ce schéma, les auteurs ont demandé à un panel de personnes de comparer visuellement la version originale du document et sa version tatouée. Personne n'a pu différencier les deux documents. Cependant, nous pouvons nous interroger sur l'intérêt d'une telle expérience. Elle pourrait avoir du sens en tatouage d'image où la perception du document est une notion importante. Or, les documents géographiques sont aussi conçus pour être utilisés par des algorithmes de géomatique et pas directement par des humains. Il aurait donc été plus intéressant de vérifier si l'exécution de tels algorithmes est modifiée par le tatouage.

M. Voigt *et al.* ont développé dans [Voigt et Busch, 2002] une méthode de tatouage basée sur la modification de chiffres significatifs dans les coordonnées des sommets du document. Ce schéma propose de considérer seulement deux de ces chiffres choisis en fonction de la précision du document original. Ici, on considére le document dans son intégralité, il n'est pas question de le découper en différentes parties. Les sommets sont ordonnés et pour chacune de leurs coordonnées, les chiffres significatifs sont modifiés de façon à insérer un bit d'une séquence pseudo-aléatoire. La clé constitue la graine de cette séquence. Les modifications sont effectuées en garantissant que le déplacement de chaque sommet ne peut dépasser la précision initiale du document. Ce schéma est aveugle et le déplacement des sommets est borné. De plus, le schéma résiste à l'ajout de bruit. Par contre, les transformations qui consistent à ajouter, supprimer des sommets ou à découper le document ne sont pas étudiées dans l'article. On remarque néanmoins que celles-ci vont perturber la synchronisation du générateur pseudo-aléatoire. On peut donc supposer que l'algorithme de détection n'est pas robuste à ces transformations.

À la suite de ce travail, les même auteurs ont proposé une autre méthode basée sur l'aspect aléatoire des documents géographiques [Voigt et Busch, 2003]. Les auteurs mettent en évidence une certaine propriété des sommets d'un document géographique. Dans un document quelconque,

les sommets suivent cette propriété avec une certaine distribution. Le tatouage consiste à modifier cette distribution pour certains sommets sélectionnés. La sélection des sommets est effectuée en deux temps. Tout d'abord, une région de la carte est sélectionnée afin de porter la marque. Cette région est ensuite découpée en un ensemble de « patches ». Pour un « patch » donné, une séquence pseudo-aléatoire sert à répartir les « patches » en deux groupes A et B. Chaque « patch » est ensuite découpé en un ensemble de petits rectangles. Selon que le « patch » appartient au groupe A ou B, les sommets du rectangle sont déplacés différemment. Dans un document non tatoué, la différence de variance des positions des sommets entre les groupes A et B est faible. Par contre, après tatouage, celle-ci est augmentée. Pour détecter la marque, on effectue les opérations de découpage du document, puis on calcule le ratio entre la variance des positions des sommets de chacun des deux groupes. Lorsque ce ratio est inférieur à un certain seuil, on considére le document non-tatoué. Sinon, on considére qu'il est tatoué. Le schéma est robuste contre un ensemble de transformations : interpolation de sommets, simplification de sommets, ajout de bruit et dans une certaine mesure, découpage du document. De plus, le schéma est aveugle et le déplacement des sommets est borné. Par contre, l'utilisation du générateur pseudo-aléatoire peut rendre le schéma sensible à une désynchronisation. De plus, le schéma permet seulement de borner le déplacement des sommets. À aucun moment, il ne permet de garantir que le tatouage ne va pas modifier une autre propriété du document, comme sa topologie.

Un troisième schéma a été proposé par M. Voigt et G. Schulz [Schulz et Voigt, 2004] afin d'inclure un message dans un document géographique. Le document est quadrillé en carrés dont la diagonale est égale à quatre tiers de la précision du document. Afin de marquer un 0 ou un 1, on déplace les sommets d'un côté ou de l'autre du carré en suivant un des deux axes (horizontal ou vertical). Le déplacement d'un sommet sur chaque axe est donc borné par la précision du document. Le schéma pose des problèmes de pertes d'information dues à certaines transformations et de synchronisation lors de la détection. Les auteurs proposent des solutions basées sur les codes correcteurs et des patrons de synchronisation. Le schéma est aveugle et code un message dans le document. Le message peut servir à incorporer les méta-informations sur le document au sein même de ses géométries. Ainsi, les meta-informations sont conservées même si l'on change le format de stockage du document. Le schéma résiste dans une certaine mesure à l'ajout de bruit, à la suppression et à l'ajout de sommets. De plus, les auteurs proposent une légère adaptation du schéma afin de le rendre résistant au découpage du document ou à l'algorithme de simplification de Douglas-Peucker. Le découpage consiste à conserver uniquement les sommets dans un rectangle. La simplification de Douglas-Peucker supprime les sommets d'une polyligne ou d'un polygone qui sont le moins significatifs mais ne déplace pas de sommets. Ces deux transformations sont couramment utilisées par les algorithmes de géomatique. Dans ce schéma, les auteurs ne contrôlent pas la déformation impliquée par l'algorithme de tatouage.

Terminons par le schéma de J. Lafaye, J. Béguec, D. Gross-Amblard et A. Ruas spécialement conçu pour tatouer la couche bâti. Le schéma est détaillé dans [Lafaye, 2007], [Lafaye *et al.*, 2007a], [Lafaye *et al.*, 2007c], [Lafaye *et al.*, 2007d] et [Lafaye *et al.*, 2007b]. La couche bâti représente un ensemble de bâtiments sous la forme de polygones. Le schéma tient compte de la

particularité de ce type de données afin de proposer un schéma robuste à l'équarissage des polygones. Cette transformation consiste à simplifier un polygone en renforçant ses angles droits, ce qui est particulièrement adapté pour le bâti. Les auteurs utilisent l'orientation principale du polygone qui est la somme pondérée de l'orientation de ses côtés par leur longueur. Ils ont montré que la longueur du polygone suivant son orientation principale n'est que peu influencée par l'équarissage. La subtilité de l'algorithme est d'allonger les polygones le long de cette orientation principale. L'algorithme de tatouage est largement inspiré des travaux de D. Gross-Amblard [Gross-Amblard, 2003] sur le tatouage de base de données relationnelles. Afin de déterminer quels polygones doivent être allongés, les auteurs utilisent un identifiant robuste, calculé à partir des bits de poids fort du centroïde du polygone, et une clé secrète. Pour chaque polygone sélectionné, l'algorithme effectue une élongation selon l'orientation principale du polygone. La nouvelle longueur du polygone est calculée en fonction de l'identifiant du polygone, de la clé secrète et d'un intervalle de quantification paramétré par l'utilisateur. Celui-ci permet de borner le déplacement des sommets. L'élongation des polygones peut entraîner une superposition de certains polygones du document. Lorsque ce cas survient, la modification du polygone est abandonnée. La détection se fait de manière statistique. Dans un document tatoué, la plupart des polygones sélectionnés vont avoir une longueur biaisée. En reconstituant la sélection avec la même clé que lors du tatouage, on évalue la proportion de polygones qui possédent un biais. Lorsque cette proportion est trop peu probable (i.e. dépasse un seuil), on admet que le document est tatoué. Le schéma est aveugle. Il est robuste à l'équarissage, la suppression d'un certain nombre de polygones, l'ajout de polygones, la simplification des polygones. Le principal apport de ce schéma vient du fait que les auteurs se sont attachés, plus que les autres, à tenir compte de la particularité des données géographiques ainsi que de l'utilisation de ce type de données. Au final, ils ont produit un schéma particulièrement bien adapté au tatouage de la couche de bâti. On notera que toutes les étapes sont effectuées localement au polygone ce qui rend le schéma résistant au découpage.

Les schémas basés sur l'ajout de sommets

Certains schémas préservent les positions des sommets. Ils se contentent d'ajouter des sommets le long des arêtes. B. Huber a proposé un schéma [Huber, 2002] utilisant ce principe et une mise en œuvre pour ArcView, le logiciel de visualisation et de traitement de données géographiques d'ESRI. L'algorithme de tatouage sélectionne l'arête la plus longue du polygone. Des sommets sont ajoutés par interpolation le long de l'arête. La distance entre deux sommets consécutifs d'une polyligne permet de coder un bit. La distance qui permet de coder un bit à 1 est paramétrée par l'utilisateur. Pour coder un bit à 0, on prend la moitié de cette distance. L'algorithme de détection est trivial. On peut noter que l'algorithme assure la confidentialité du message grâce à un chiffrement par une clé secrète. Des codes correcteurs d'erreurs permettent de retrouver le message si celui-ci a été un peu altéré. Ce schéma est aveugle et permet d'inclure un message dans le document. Il est robuste à la translation et la rotation. Par contre, un simple algorithme de simplification va pouvoir faire disparaître la marque. Les sommets ajoutés n'ont

pas de sens particulier pour la donnée, ils peuvent donc être supprimés sans que cela nuise à la qualité du document. De plus, le codage d'un bit implique l'ajout de sommets. Lors du tatouage, le document est donc amené à grossir. D'une part, cela augmente les coûts liés au stockage et à la diffusion, d'autre part, le traitement du document devenu plus gros sera plus lent.

D'autres schémas plus ou moins similaires ont été présentés par d'autres auteurs. H. Sonnet *et al.* [Sonnet *et al.*, 2003] proposent de poser 8 sommets par segment, à égale distance les uns des autres. La présence ou non d'un de ces sommets permet de coder un bit. Il code ainsi 8 bits par segments. K.T. Park *et al.* [Park *et al.*, 2002], donnent une autre variante qui consiste à découper le document en rectangles. Les auteurs ajoutent des sommets en utilisant les coefficients fréquentiels de ces rectangles. Ces méthodes présentent les mêmes inconvénients que ceux de la méthode de B. Huber.

3.2.4 Étude des principaux aspects du tatouage de documents géographiques

Cette section fait le bilan des schémas présentés ci-avant afin d'en étudier les différents aspects. Dans un premier temps, nous discutons la notion de document géographique et de qualité. Ensuite, nous abordons le sujet des transformations qui sont susceptibles d'être appliquées à un document géographique. Enfin, nous montrerons les avantages des schémas aveugles pour les documents géographiques vectoriels.

La notion de document géographique

Dans les travaux précédemment cités, un document géographique est vu soit comme un ensemble de géométries (essentiellement des polygones), soit comme un graphe où les sommets sont étiquetés par leurs coordonnées. Il est facile de passer d'un ensemble de géométries à un graphe. En revanche, il est difficile d'opérer la transformation inverse de façon fiable. Les schémas qui traitent des ensembles de géométries ont donc accès à une information plus riche. Considérons les cas de figure suivants :

- on tatoue un ensemble de polygones ;
- on tatoue un ensemble de polylignes ;

Tant qu'il s'agit de tatouer des polygones, par exemple des bâtiments, il est tout à fait possible de travailler avec des ensemble de géométries ou avec un graphe. Cependant, travailler au niveau des géométries semble mieux adapté. Dans le cas contraire, on perd par exemple la notion d'intérieur et d'extérieur. D'autre part, on voit difficilement dans quel cas un utilisateur a intérêt à transformer un ensemble de polygones en graphe.

En revanche, lorsque l'on veut tatouer des polylignes, comme des fleuves ou des routes par exemple, on a intérêt à travailler sur un graphe. En effet, les polylignes peuvent être découpées ou recollées afin de produire un document dont les géométries sont complétement différentes, sans perdre d'information. Par coupage et collage des polylignes, on peut obtenir une multitude de documents. Cependant, on peut noter que le graphe issu du document transformé demeure identique à celui issu du document de départ.

On peut donc conclure que si l'on souhaite concevoir une méthode de tatouage applicable aux polylignes ou à la fois aux polygones et aux polylignes, on a tout intérêt à considérer les données comme un graphe.

La notion de qualité

Lorsqu'on exclut les rares méthodes qui ajoutent des sommets dans les documents, les schémas que nous avons vus tatouent par déplacement des sommets. Les différents schémas ont des points de vues très différents concernant les implications de ces déplacements sur la qualité du document. La norme ISO 8402-94 définit la qualité ainsi : « Ensemble des caractéristiques d'une entité qui lui confèrent l'aptitude à satisfaire des besoins exprimés et implicites. »

Nous avons vu que certains auteurs utilisent le PSNR pour mesurer l'impact des modifications sur le document. D'autres utilisent un panel d'utilisateurs humains qui vont décider si le document semble avoir été modifié ou non. De telles mesures ont un sens pour les images car elles sont lues, perçues et traitées par des humains. Cependant, cela n'a pas de sens lorsque l'on traite des données géographiques. En effet, ce type de données a pour but d'être lu et traité par des programmes informatiques. Il est donc plus intéressant de vérifier si le résultat de l'exécution de ces programmes est modifié par l'application de la marque. Ainsi, il vaut mieux s'intéresser à respecter un certain nombre de contraintes sur le document. Les contraintes peuvent être métriques, pour borner le déplacement de certains éléments du document mais elles peuvent aussi être topologiques, pour empêcher de changer la relation entre certains éléments du document.

La contrainte métrique la plus souvent traitée dans la littérature est la garantie de borner le déplacement des sommets. Un document géographique est donné avec une certaine précision qui représente l'écart maximal entre les coordonnées des sommets du document et leurs positions réelles. X. Niu *et al.* [Niu *et al.*, 2006] affirment que cette précision donne l'amplitude maximum autorisée pour les déplacements et que rester en deçà de ce seuil ne va pas dégrader la validité du document. Nous préférons adopter un autre point de vue. Pour nous, le déplacement des sommets va forcément impliquer une perte de la précision du document. Il faut pouvoir borner l'amplitude du déplacement des sommets afin de pouvoir contrôler a priori la perte de précision autorisée.

Les contraintes topologiques ne sont que peu traitées dans la littérature. J. Lafaye *et al.* dans [Lafaye *et al.*, 2007a] sont les seuls qui proposent d'empêcher l'algorithme de tatouage de créer des collisions entre les polygones : l'algorithme de tatouage abandonne toute modification qui impliquerait une violation de cette contrainte topologique.

La littérature n'aborde pas le problème des contraintes d'orientation qui serviraient, par exemple, à forcer un sommet à demeurer à l'est ou à l'ouest d'un autre.

Les transformations légitimes

Les schémas que nous avons présentés sont robustes dans le sens qu'ils résistent à un ensemble de transformations. Globalement, les auteurs ne font pas vraiment de distinction entre tranformation légitime et attaque volontaire. Les transformations habituellement considérées sont regroupées en 5 classes par X. Niu *et al.* [Niu *et al.*, 2006] :

- transformations géométriques (translation, rotation, etc.) ;
- suppression de sommets ;
- ajout de sommets ;
- réordonnancement des données ;
- bruitage de faible amplitude.

Cette classification est largement inspirée de celle effectuée initialement par M. Kutter *et al.* dans [Kutter et Petitcolas, 1999] pour le tatouage d'images et reprise par C. Lopez [Lopez, 2002]. Elles doivent être prises en compte avec précaution lorsque l'on travaille sur les données géographiques. Dans cette section, nous étudions chacune de ces transformations. Nous présentons aussi une transformation supplémentaire : l'« overmarking » proposée par [Lopez, 2002]. Cette transformation consiste à appliquer l'algorithme de tatouage avec une autre clé.

Transformations géométriques La robustesse du schéma face aux transformations géométriques est importante. Sans cette robustesse, un utilisateur peut laver la marque en opérant par exemple une légère rotation. Une telle transformation s'effectue facilement. De plus, elle ne nuit ni à la topologie ni à la métrique du document.

Suppression de sommets On peut imaginer que l'utilisateur va vouloir appliquer une simplification de ces données initiales pour en faciliter le traitement. Par exemple, il peut utiliser l'algorithme de Douglas-Peucker [Douglas et Peucker, 1973a]. Cependant, ce type d'algorithme va diminuer la qualité du document.

On peut aussi considérer le découpage (*cropping*) comme une transformation potentielle. Dans ce cas, les sommets et les arêtes qui sont en dehors d'un rectangle choisi sont supprimés. À l'intérieur de la zone sélectionnée, le document n'est pas altéré.

La plupart des schémas de tatouage de documents géographiques considèrent la suppression aléatoire de sommets comme une transformation possible. Cependant, il est peu probable que la suppression de sommets soit effectuée de façon aléatoire par un utilisateur. En effet, une telle transformation aurait trop de chances de mener à un document incohérent car un sommet très significatif a autant de chances d'être supprimé qu'un sommet qui l'est moins.

Ajout de sommets Ajouter des sommets aléatoirement le long des arêtes n'est pas très intéressant pour un utilisateur. En effet, cela produit un document plus lourd. De plus, une étape de filtrage avant la détection peut facilement supprimer ces sommets. C'est cependant une transformation souvent considérée par les schémas de tatouage de données géographiques.

Le réordonnancement des données Le réordonnancement des données regroupe plusieurs transformations. Parfois, Il s'agit de modifier l'ordre dans lequel sont stockés les polygones. On peut aussi changer l'ordre dans lequel sont énumérés les sommets des polylignes et des polygones. La liste des sommets du polygone étant cyclique, on peut ainsi changer le premier sommet du polygone.

L'ajout de bruit gaussien L'ajout de bruit gaussien est souvent utilisée en tatouage d'images. Toutefois, il semble étrange de considérer l'ajout d'un bruit gaussien comme une transformation potentielle pour les données géographiques. En effet, un utilisateur a tout intérêt à conserver la qualité des informations fournies. L'ajout d'un bruit gaussien va à la fois diminuer la précision du document et risquer d'ajouter des erreurs topologiques. Par exemple, deux routes parallèles risquent de devenir sécantes.

Le re-tatouage (*overmarking* en anglais) Cette transformation consiste à appliquer un tatouage par dessus un document déjà tatoué. Elle n'est pas considérée dans les schémas que nous avons présentés. Cependant, quand le schéma de tatouage est public, on peut facilement imaginer qu'un attaquant puisse utiliser cette transformation pour laver le document tatoué.

Les propriétés des schémas de tatouage

Dans cette partie, nous nous intéressons aux propriétés que possèdent les algorithmes de tatouage de données géographiques. Notre objectif est de proposer un schéma de tatouage de documents géographiques dans le cadre de la protection de la propriété intellectuelle. Nous nous sommes donc restreints à présenter des travaux concernant le tatouage robuste.

Tatouage aveugle ou non Certains des schémas ne sont pas aveugles. Par définition, ils nécessitent de conserver le document original afin de pouvoir l'utiliser lors de la détection. Cela pose plusieurs problèmes. Tout d'abord, pour prouver que le document nous appartient, il faut pouvoir donner un couple composé du document original et de la clé. Or, dans les schémas non-aveugles que nous avons étudiés, un attaquant peut créer un tel couple à partir du document tatoué. La seule façon d'éviter ce problème est de faire appel à un tiers de confiance pour archiver le document original. L'archivage peut alors s'avérer coûteux car les documents géographiques sont souvent lourds et pour chaque document publié, il faut conserver l'original en lieu sûr. Les schémas aveugles ne présentent pas ces problèmes d'archivage.

Tatouage réversible ou non Les schémas de M. Voigt *et al.* [Voigt *et al.*, 2004] et de B. Huber *et al.* [Huber, 2002] sont réversibles. On peut s'interroger sur les cas d'utilisation pour lesquels ce type de schéma est utile. Selon M. Voigt *et al.* , cette application est utile pour tatouer des données militaires. Cela permet à un utilisateur de retrouver le document dans sa précision originale en cas de besoin. Cependant, si l'on permet à un client d'effectuer cette opération, il dispose alors du document original qu'il peut diffuser. Pour empêcher cela, il faut intégrer

l'algorithme de reconstruction du document original au plus près du GIS, éventuellement en mettant en jeu des méchanismes complexes de DRM [3]. On obtient le même résultat en publiant d'une part le document tatoué et d'autre part le document original avec une DRM dans le cas d'une application militaire. On peut ainsi contrôler les clients auxquels on fournit les données originales. Donc, la réversibilité d'un schéma de tatouage de données géographiques ne parait pas très pertinente.

Tatouage par transformée ou par modification géométriques Les méthodes de tatouage par transformées garantissent que l'aspect des géométries sera préservé lors de la phase de tatouage. Par contre, elles ne permettent pas de contrôler le déplacement des sommets du document. On ne peut donc pas facilement borner la perte de précision due au tatouage. Les méthodes utilisant des déplacements de sommets paraissent donc mieux adaptées pour tatouer des données géographiques.

Conclusion

Dans ce chapitre, nous avons présenté différents schémas de tatouage de données géographiques et avons dégagés les principaux aspects de ces schémas. On constate que peu des schémas présentés tiennent réellement compte de la spécificité des documents géographiques vectoriels. Généralement ils bornent le déplacement des sommets mais peu s'intéressent à la préservation de la topologie du document qui est pourtant un aspect important de la donnée géographique. D'autre part, les auteurs construisent des schémas robustes à des transformations sans réel intérêt pour les données géographiques.

[3] Digital Right Managment

Deuxième partie

Tatouage de documents géographiques

Chapitre 4

Présentation de notre schéma de tatouage

Sommaire

Dans ce chapitre, nous présentons le schéma de tatouage que nous avons conçu pour le documents géographiques vectoriels. Ce schéma a été élaboré dans le cadre du projet Tadorne (Tatouage de Données Contraintes) de l'ACI Sécurité et Informatique auquel participaient les laboratoires du GREYC, du Cédric (CNAM, Paris), du Lamsade (Paris-Dauphine), du Le2i (Université de Bourgogne) et enfin le laboratoire Cogit de l'IGN (Institut National Géographique). Le schéma propose une méthode de tatouage pour les données géographiques vectorielles particulièrement adaptée aux documents géographiques représentant des données routières. Le schéma garantit de préserver certaines qualités topologiques et métriques du document lors de la phase de tatouage.

Dans un premier temps, nous présentons les données géographiques considérées ainsi que les idées directrices de notre schéma. Nous détaillons enfin le schéma que nous avons conçu et précisant chacune de ces étapes.

4.1 Domaine d'application et cadre de travail

Cette section va préciser les aspects de notre schéma, notamment en termes de robustesse et de qualité à préserver, en gardant à l'esprit les spécificités des documents géographiques vectoriels.

4.1.1 Données géographiques considérées

Notre objectif est de proposer un schéma de tatouage de données géographiques vectorielles. Bien qu'il puisse exister des informations descriptives associées aux objets géographiques, nous utilisons uniquement les données géométriques car nous estimons que ces données sont suffisamment riches pour y insérer une marque. De plus, cela permet d'être totalement indépendant de la présence, de l'absence ou de la modification des données alphanumériques.

Nous travaillons sur le graphe extrait des données géographiques. Dans le cas des données routières, les sommets du graphe représentent les virages ou les croisements des routes et les arêtes sont des portions de routes. Chaque sommet du graphe est étiqueté par ses coordonnées.

4.1.2 Préservation de la qualité

Nous souhaitons que l'algorithme de tatouage préserve la qualité du document original. Cette notion de préservation de qualité est centrale pour le schéma de tatouage que nous avons construit car c'est la qualité de l'information qui lui confère sa valeur. Nous devons donc garantir que la dégradation du document engendrée par l'algorithme de tatouage sera contrôlée et bornée.

Notre approche consiste à préserver cette qualité à tout moment pendant le processus de tatouage. Pour cela, nous définissons la notion de préservation de qualité pour les documents géographiques vectoriels. Rappelons que la préservation de qualité est représenté par une relation $Q_{\mathcal{D}}$ et qu'un algorithme de tatouage préserve la qualité des documents si cette relation est vérifiée

entre tout document et son tatoué. La relation $Q_{\mathcal{D}}$ pour les données géographiques vectorielles fait intervenir des notions de topologie et de métrique.

Précision du document

Nous voulons borner le déplacement de chaque sommet du document. Cette borne est un paramètre de notre algorithme choisi par l'utilisateur. La donnée géographique originale est fournie avec une précision qui indique la distance maximum entre la position réelle et la position représentée des sommets du document. Le déplacement des sommets va engendrer une perte de précision de la donnée géographique. En bornant le déplacement des sommets, nous pouvons garantir, à priori, la valeur de la précision après tatouage.

Les sommets et les arêtes du document

Nous voulons conserver chaque sommet et chaque arête présent dans le document original. En effet, la suppression d'une arête du graphe reviendrait par exemple à enlever une route entre deux points. L'ajout d'une arête reviendrait à créer un tronçon de route qui n'existe pas dans la réalité. On imagine aisément l'impact que ce genre de modification pourrait avoir sur les algorithmes qui traitent le document.

Préservation de la triangulation

Nous inspirant des travaux de R. Ohbuchi *et al.* , nous utilisons la triangulation de Delaunay de l'ensemble des sommets du document. Notre schéma garantit que celle-ci reste inchangée avant et après tatouage. Plus formellement, toute paire de sommets qui sont reliées dans la triangulation originale le restera dans la triangulation du document tatoué. Nous posons l'hypothèse raisonnable qu'en préservant la triangulation de Delaunay, nous modifions très peu la topologie du document.

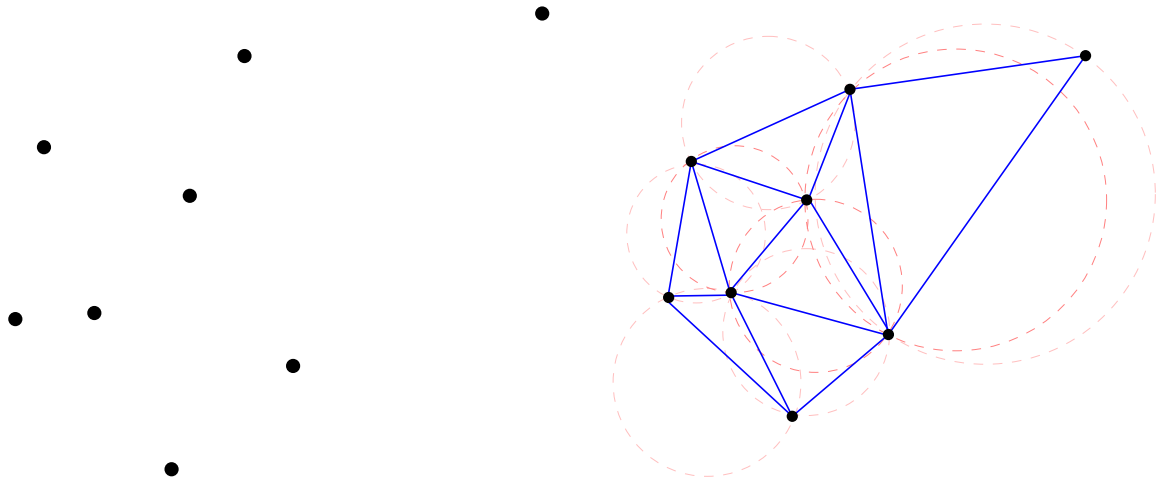
La triangulation de Delaunay sur un nuage de points permet d'obtenir une triangulation telle que tout triplet de sommets forme un triangle si et seulement si et seulement si aucun sommet n'est à l'intérieur du cercle circonscrit à ce triangle. Par construction, cette triangulation maximise l'angle minimum des triangles et elle favorise l'équilatéralité des triangles.

La figure 4.1(a) présente un nuage de points et la figure 4.1(b) donne la triangulation de Delaunay qui lui est associée. On peut vérifier qu'aucun sommet n'est à l'intérieur du cercle circonscrit à l'un des triangles composant la triangulation.

Notons que les trois sommets qui forment le triangle sont sur le cercle circonscrit est non à l'intérieur.

Bilan de la préservation de qualité

La relation de préservation de qualité ne sera pas vérifiée entre un document et son tatoué dans les cas suivants :



(a) Nuage de points (b) Triangulation de Delaunay du nuage de points

FIG. 4.1 – Exemple de triangulation de Delaunay

– un sommet a été déplacé d'une distance supérieure à la perte de précision autorisé ;
– un sommet ou une arête a été ajoutée ou supprimée ;
– la triangulation de Delaunay du nuage de point du document est modifiée.

Dans les autres cas, la relation sera vérifiée.

4.1.3 Triangulation de Delaunay

Les exemples de la figure 4.2 illustrent une triangulation de Delaunay associée à un graphe issu de données routières réelles. Les figures 4.2(a) et 4.2(b) représentent respectivement un réseau routier et son nuage de points associé. Enfin, la triangulation de Delaunay du nuage de points est donnée par la figure 4.2(c). Comme le montre cet exemple, tous les triangles sont inscrits dans un polygone qui représente l'enveloppe extérieure de la triangulation de Delaunay et que l'on appelle l'enveloppe convexe. Il représente le polygone convexe minimum qui contient tous les sommets de la triangulation.

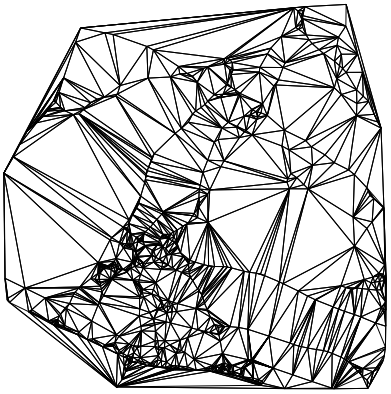
On peut noter que pour un nuage de points donné, la triangulation de Delaunay associée n'est pas forcément unique. Par exemple, si les quatre sommets de deux triangles adjacents sont cocycliques, alors il existe deux façons de découper cet ensemble de quatre sommets en deux triangles. Cependant, il s'agit d'un cas pathologique qui ne nous gênera pas dans nos applications et nous allons considérer que la triangulation sur laquelle nous travaillons est unique. D'ailleurs, un logiciel calculant la triangulation peut lever cette ambiguïté pour produire une triangulation déterministe.

4.1.4 Modèle de l'utilisateur

Nous avons modélisé un utilisateur comme étant capable d'effectuer un ensemble fixé de transformations sur le document avant de revendre ou redistribuer celui-ci. Le schéma que nous

(a) Document original (Source IGN)

(b) Nuage de points associé



(c) Triangulation de Delaunay

FIG. 4.2 – Exemple de triangulation de Delaunay sur des données réelles.

présentons doit être résistant à ces transformations, au moins dans une certaine mesure que les expériences permettront de quantifier. Nous nous sommes intéressés aux transformations suivantes :

- la réorganisation des données ;
- le découpage de la carte ;
- les transformations géométriques : rotation ou translation ;
- le retatouage.

Dans cette section, nous allons expliquer en quoi consistent ces opérations et pourquoi nous les avons choisies.

Réorganisation des données

La réorganisation des données est une transformation potentielle que nous prenons en compte. Il est possible de définir cette transformation de plusieurs façons. Pour notre part, nous considérons n'importe quelle transformation pour laquelle le graphe extrait du document original et celui

extrait du document transformé sont identiques. Une telle transformation peut impliquer une grande perte d'information. Toutefois elle est utile dans le cas, par exemple, d'un changement de format du document. De plus, on ne peut garantir qu'un algorithme de géomatique va préserver l'ordre dans lequel sont stockés les objets.

Découpage de la carte

Une transformation particulièrement pertinente est le découpage du document. Elle consiste à filtrer tous les sommets et les arêtes qui ne sont pas inclus dans un rectangle donné. Le découpage est une transformation naturelle qui est utilisée pour isoler la partie des données qui intéresse l'utilisateur. Elle permet par exemple d'isoler une ville dans un ensemble de données représentant un pays ou une région.

La figure 4.4 présente un exemple de découpage d'un document. Le document original est représenté par la figure 4.3(a). Dans cette figure, le rectangle foncé représente la zone découpée. La figure 4.3(b) illustre la zone découpée.

Transformations géométriques

Nous considérons aussi les transformations géométriques telles que la rotation des données ou leur translation. Bien que ces transformations ne soient pas les plus utilisées, la plupart des méthodes de tatouage présentées dans la littérature [Niu *et al.*, 2006] les considèrent comme des transformations légitimes. Cela s'explique essentiellement par le fait qu'il est facile de laver une marque qui ne résisterait pas à ces modifications.

4.1.5 Schéma aveugle

Nous proposons un schéma aveugle. Il n'est donc pas nécessaire de connaître le document original pour que l'algorithme de détection vérifie si une marque est présente dans un document. En utilisant ce genre de schéma, nous disposons de moins d'information pour décider si une marque est présente dans un document donné.

Cependant, un schéma de tatouage aveugle est préférable car les données géographiques peuvent être volumineuses et sujettes à des mises à jour. Un schéma non-aveugle demanderait de stocker tout document original qui a permis de créer un document tatoué. De plus, essayer de détecter une marque dans un document particulier demanderait de le comparer à chaque document original sauvegardé. Enfin, pour chaque test de détection, il faut divulguer le document original non-tatoué à un tiers, par exemple à l'expert indépendant qui doit effectuer le test de détection pour un tribunal. Pour toutes ces raisons, et bien que les schémas aveugles soient plus difficiles à concevoir, nous avons concentré nos efforts vers la construction de tels schémas.

(a) Document original

(b) Document découpé

FIG. 4.3 – Exemple de découpage.

4.1.6 Schéma 0-bit

Pour résister au découpage du document, nous avons construit un schéma qui ne pose pas de problème de synchronisation. Au contraire d'un schéma n-bits, qui consiste à cacher un message au sein du document, un schéma zéro bit permet juste de répondre à la question : « Le document

est-il tatoué avec cette clé ». Les schémas 0-bit sont donc moins spectaculaires car aucun message n'apparaît lors de la détection mais ils s'avèrent tout aussi efficaces. Pour les schémas aveugles n-bits, le principal problème est de resynchroniser l'algorithme de détection sur le message caché. Notre schéma est de type zéro-bit, c'est à dire qu'aucun message n'est inséré dans le document. Nous introduisons un biais statistique en fonction d'une clé. Sachant que ce biais a une chance extrémement faible d'être présent dans un document qui n'a pas été tatoué par ladite clé, lorsque nous observons le biais, nous concluons que la clé a été utilisée pour tatouer le document. Nous évitons ainsi tout problème de synchronisation.

Conclusion

Nous avons présenté les aspects sur lesquels nous avons conçu notre schéma. Nous souhaitons construire un schéma zéro-bit, aveugle, robuste à la réorganisation des données, aux transformations géométriques et au découpage du document et au retatouage. L'aspect le plus important du schéma est de tenir compte à la fois de la topologie du document et de sa métrique. Le schéma va à la fois borner le déplacement des sommets du document et préserver la triangulation de Delaunay du document. Dans la section suivante, nous verrons comment notre approche locale permet de tenir compte de tous ces aspects.

4.2 Idées directrices

Cette section présente les idées directrices de notre schéma. Afin de garantir une bonne résistance de la marque vis-à-vis du découpage du document et d'obtenir un schéma aveugle, nous avons privilégié une approche locale. Nous travaillons donc sur de petites parties significatives extraites du document original que nous nommons *sites*.

Pour chaque site, nous calculons un identifiant robuste, que nous appelons *codage*. Un sous-ensemble des sites du document est sélectionné en fonction de ce codage et d'une clé secrète. Les sites sélectionnés sont ensuite forcés à satisfaire ou non une propriété Φ dont nous spécifierons le rôle. Pour forcer un site à satisfaire Φ nous modifions celui-ci. Pour garantir que ces modifications n'entraîneront pas de perte de qualité du document, nous définissons une notion de qualité locale au site. Nous verrons que respecter cette qualité locale suffit à conserver la qualité globale du document.

4.2.1 L'approche locale

Nous souhaitons créer un schéma aveugle qui opère localement pour garantir une bonne résistance au découpage du document. Notons que les méthodes de tatouage qui considèrent le document de façon globale requièrent le document complet pour que la détection puisse être effectuée. Elles sont donc plus sensibles au découpage du document.

Notre idée est d'extraire, de marquer puis de réintroduire, une par une, de petites parties du document que nous nommons *sites*. Dans le cas des données géographiques, un site sera composé

d'un sous-ensemble de sommets et d'arêtes du graphe original. Une définition complète de la notion de site sera donnée dans la section 4.3.1.

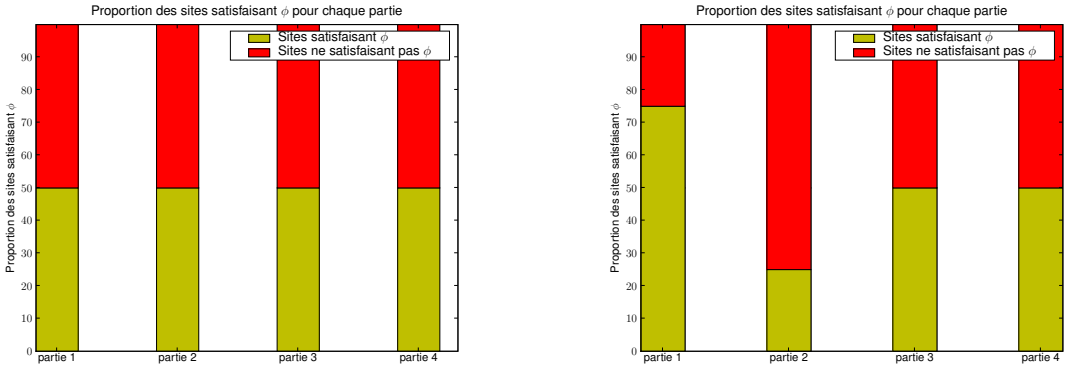
4.2.2 Préservation locale de la qualité

Le schéma de tatouage présenté préserve la qualité du document tout au long du processus de tatouage. Afin de travailler localement sur chaque site, notamment pour les forcer à satisfaire ou non Φ, nous définissons une notion de préservation de qualité locale. Nous avons défini la qualité locale de sorte que si l'on ne modifie qu'un seul site en préservant sa qualité, la qualité globale du document est préservée. Ainsi, l'algorithme de tatouage va travailler itérativement sur les sites du document en n'appliquant les modifications des sites sur le document uniquement quand celle-ci préserve la qualité du site. La section 4.3.2 détaillera cette notion de préservation de qualité pour les sites.

4.2.3 Les aspects du site

Pour marquer un document, nous partitionnerons l'ensemble des sites qu'il contient pour introduire un biais statistique dans les deux premières parties. Le partitionnement est obtenu par un codage des sites et la clé. Le biais portera sur la proportion des sites qui satisfont Φ.

La figure 4.4(a) donne un exemple de la proportion des sites qui satisfont Φ dans les différentes parties d'une partition en 4 parties dans un document non-tatoué. La figure 4.4(b) présente la même chose, cette fois-ci pour un document tatoué.



(a) document *non-tatoué*.

(b) document *tatoué*.

FIG. 4.4 – Exemples de la proportion de sites qui satisfont Φ pour deux documents, l'un est tatoué, l'autre non. Les deux documents sont partitionnés en 4 parties.

Le codage d'un site

Pour chaque site, nous avons besoin d'un codage robuste qui servira, avec la clé secrète, à sélectionner le sous-ensemble des sites du document qui seront modifiés pour porter la marque.

Nous verrons dans la partie 4.4.2 que le codage n'utilise que des informations locales au site qui sont basées sur les critéres topologiques et que nous le choisissons pour être robuste notamment aux transformations géométriques. Notons bien qu'il est nécessaire d'avoir un nombre de codages différents assez important. Nous n'imposons pas au codage d'être unique, mais nous souhaitons minimiser les collisions. Ces collisions forment des classes de sites de même codage qui seront tous traités de la même façon par l'algorithme. Cela peut donner des indices pour un attaquant voulant laver la marque. Nous discuterons de ces problèmes dans la partie expérimentation.

Le partitionnement des sites

Afin de sélectionner certains sites pour porter la marque, nous partitionnons les sites extraits du document original selon leur codage et la clé secrète. Le nombre de parties p (où $p \geq 2$) est un paramètre du schéma de tatouage qui permet de choisir la proportion de sites modifiés lors du tatouage. En effet, seuls les sites appartenant aux deux premières parties seront modifiés lors du tatouage. Plus on choisit p grand, plus le nombre de sites modifiés sera faible. En contrepartie, la marque sera moins présente et donc plus facile à laver. À l'inverse, plus on choisit p petit, plus la marque sera présente dans le document. Le document aura alors subit plus de modifications mais la marque sera alors plus difficile à laver. Le choix de p dépend donc de l'application choisie car il permet de régler l'intensité de la marque.

Pour deux clés différentes, nous souhaitons avoir deux partitions des sites peu corrélées. De ce fait, il est difficile pour celui qui ne possède pas la clé d'effectuer le partitionnement et donc de vérifier la présence du biais. Pour deux clés différentes, les partitions ne sont pas totalement décorrélées car deux sites de même codage appartiennent à la même partition. Dans la partie expérimentation, nous verrons comment remédier à ce problème.

La définition de la fonction de partitionnement des sites est donnée dans la partie 4.4.3. Dans la partie expérimentation, nous étudierons la distribution des sites dans les parties. Nous verrons que cette distribution n'est pas uniforme car certaines classes de sites de même codage peuvent contenir beaucoup d'éléments. Néanmoins, nous observerons que la dispersion des sites dans les parties est relativement homogène.

L'aspect aléatoire du site

Chaque site contient une partie aléatoire. Nous tirons avantage de cet aspect aléatoire afin d'introduire un biais statistique dans certaines parties de la partition. Nous définissons pour cela une propriété Φ que nous choisissons indépendante de la notion de qualité et ayant une forte probabilité de résister aux transformations du document. Dans la partie expérimentation, nous vérifierons que, pour un groupe de sites tirés aléatoirement dans un document géographique, la proportion de ceux qui satisfont Φ suit une loi normale de moyenne μ. Pour marquer un document, nous changeons significativement la proportion de sites qui satisfont Φ pour les deux premières parties de la partition. Nous donnons la définition de Φ que nous avons choisi dans la partie 4.4.4. Nous verrons qu'elle se base sur des critères métriques du site.

Conclusion

Notre schéma est basé sur la notion de *site*. Le site est une petite partie du document original pour laquelle on définit un codage et une propriété Φ sur des notions orthogonales : topologique pour le codage, métrique pour Φ.

Nous allons modifier certains sites du document en s'assurant que ces modifications préservent la qualité du site. La préservation de cette qualité de site garantit que l'on préserve la qualité globale du document.

Dans le chapitre suivant, nous donnons des définitions plus précises pour le site et la qualité de site et détaillons le fonctionnement des algorithmes de tatouage et de détection.

4.3 Présentation du schéma de tatouage

Dans cette section, nous présentons notre schéma de tatouage aveugle et robuste applicable aux documents géographiques vectoriels. Nous commencerons par expliquer la notion de site et les grandes lignes des deux algorithmes qui composent le schéma de tatouage. Rappelons que les algorithmes de tatouage et de détection prennent en entrée un graphe issu d'un document géographique dont les sommets sont étiquetés par leurs coordonnées.

Nous verrons que l'algorithme de tatouage introduit un biais statistique au sein d'un sous-ensemble des sites du document. L'introduction de ce biais est effectuée en modifiant certains sites de façon à ce qu'ils satisfassent ou non une certaine propriété Φ que nous choisissons. On observe expérimentalement que cette propriété suit une certaine distribution que nous modéliserons. En forçant certains sites à satisfaire la propriété, nous modifierons cette distribution. Rappelons que les sites modifiés sont choisis en fonction de leurs codages et de la clé secrète. L'algorithme de tatouage retourne un graphe avec les même arêtes mais dont les étiquettes sont modifiées.

Rappelons que nous garantissons que la triangulation de Delaunay des sommets du document est préservée lors du tatouage et que la distance de déplacement de chaque sommet est bornée.

L'algorithme de détection sélectionne les sites en fonction de leur codage et d'une clé secrète. Un comptage permet de déterminer le nombre de sites qui satisfont la propriété Φ. Pour décider si un document est tatoué, nous majorons la probabilité que la distribution mesurée s'écarte de la distribution modélisée. Lorsque cette probabilité franchit un seuil, nous dirons que le document est tatoué.

4.3.1 Définition des sites

Nous construisons autant de sites que le document contient de sommets. Chaque site extrait du document original est formé tout d'abord d'un sommet que nous nommons sommet central du site et de ses voisins dans la triangulation de Delaunay (ordonnés dans le sens trigonométrique). Le sommet central et ses voisins sont tous issus du document original et sont donc étiquetés par leurs coordonnées dans le plan. Celles-ci sont utilisées pour vérifier si le site satisfait la propriété

Φ que nous avons choisie. Les arêtes entre ces sommets dans le document original permettent de calculer un identifiant robuste pour le site.

Remarque 1 *La définition exacte du site nécessite aussi de connaître les sommets miroirs du sommet central du site par rapport à chacune de ses faces adjacentes dans la triangulation. Ces informations supplémentaires servent à vérifier si la qualité d'un site est altérée par un déplacement du sommet central du site.*

Nous définissons maintenant un site pour un document géographique donné par son graphe.

Définition 4.3.1 *Le site pour un document géographique est composé de quatre parties :*

- *le sommet central noté c ;*
- *ses voisins dans la triangulation notés $(n_1, n_2, ...)$;*
- *ses sommets miroirs par rapport à chacune de ses faces adjacentes dans la triangulation. Ces sommets sont notés $(m_1, m_2, ...)$;*
- *Les arêtes entre c et ses voisins ainsi que les arêtes entre les voisins, noté E_c.*

Les sommets n_i sont numérotés de façon croissante dans le sens trigonométrique. Le premier voisin, noté n_1, est choisi arbitrairement. Les sommets miroirs m_i sont numérotés de sorte que le triangle formé par les trois sommets n_i, n_{i+1} et m_i soit une face de la triangulation.

La figure 4.5 donne un exemple de site. Dans cette figure, chaque triangle représente une face dans la triangulation de Delaunay.

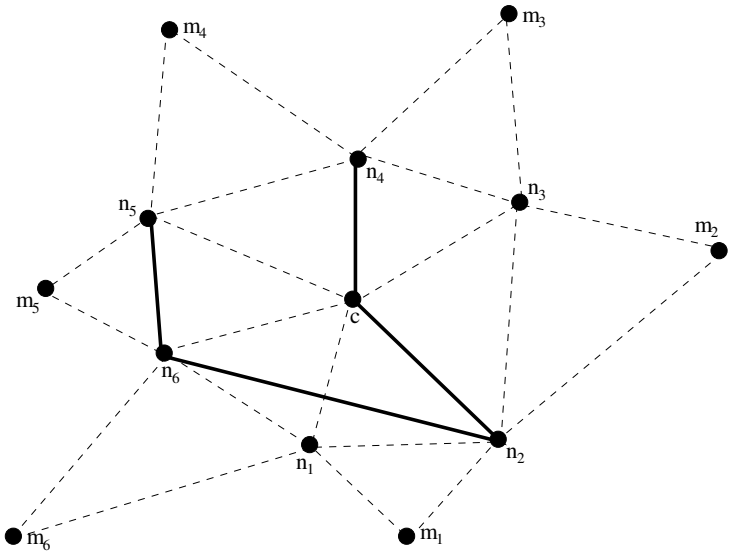
4.3.2 Préservation de la qualité des sites

La propriété Φ dépend de la position du sommet central du site par rapport à ses voisins. Pour forcer un site à satisfaire ou non Φ, nous déplaçons le sommet central du site. Nous dirons que la qualité du site est préservée après déplacement du sommet central du site, si le déplacement du sommet central est borné par la perte de précision autorisée (paramétrée par l'utilisateur) et si cette modification laisse invariante la triangulation de Delaunay du document. Vérifier si la triangulation de Delaunay est préservée après déplacement d'un des sommets est plus complexe. Ce problème de la préservation de triangulation de Delaunay pendant le mouvement des sommets de la triangulation est abordé de manière plus générale dans [Dakowicz et Gold, 2006] et [Guibas *et al.*, 1992]. Nous présentons ici une version simplifiée de ces travaux.

Afin de vérifier si la triangulation de Delaunay d'un document est préservée après déplacement d'un des sommets du document, nous devons vérifier deux choses :

- aucune face existante ne doit disparaître ;
- et aucune nouvelle face ne doit apparaître.

Pour effectuer ces deux tests, nous nous rapportons à la définition de la triangulation de Delaunay. Cette définition stipule que tout triplet de sommets dont le cercle circonscrit ne contient aucun sommet forme une face de la triangulation. Nous allons donc vérifier si la nouvelle position du sommet central le fait entrer dans le cercle circonscrit à une face auparavant vide.



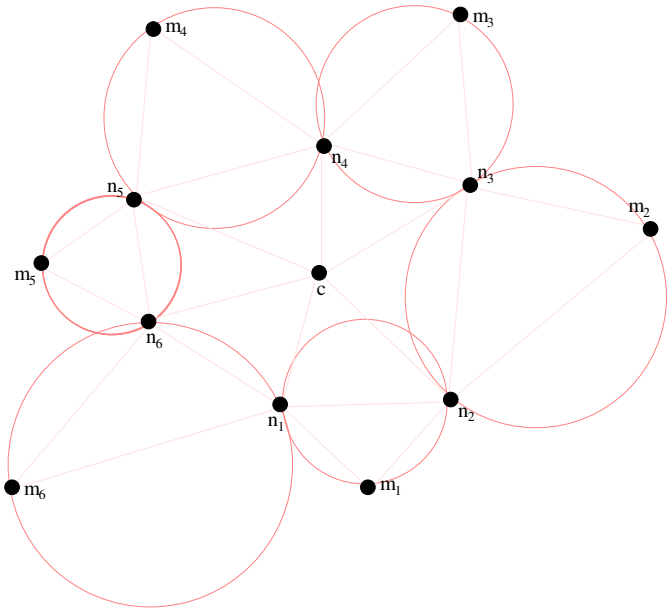
c est le sommet central du site. n_1 à n_6 sont ses voisins. m_1 à m_6 sont les sommets miroirs de c par rapport à chaque triangle adjacent à c. Les segments en gras représentent les arêtes entre c et ses voisins et les arêtes reliant les voisins entre eux dans le document original.

FIG. 4.5 – Exemple de site

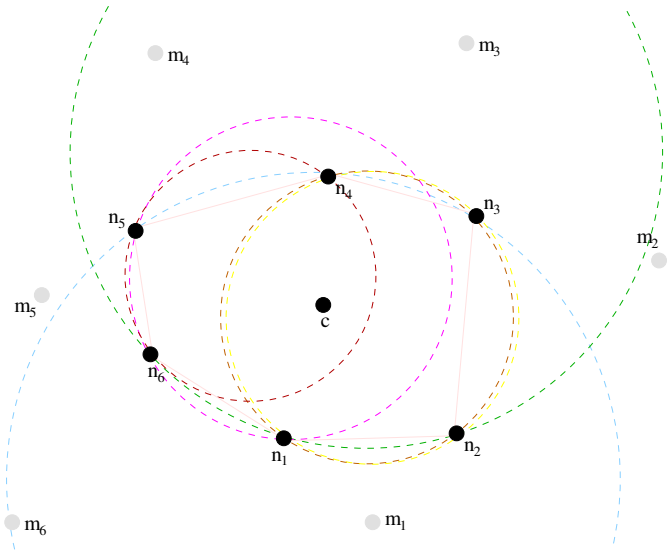
Cela voudrait dire qu'une face disparaîtrait. De plus, nous devons tester si la nouvelle position du sommet central fait apparaître un triplet de sommets dont le cercle circonscrit est vide. Cela aurait pour conséquence de faire apparaître une nouvelle face. Si l'on modifie uniquement la position du sommet central du site, sans le faire sortir du polygone de ses voisins, il est possible d'effectuer ces deux tests en utilisant seulement les données présentes dans le site.

Pour vérifier si le sommet n'entre pas dans un cercle circonscrit à l'un des triangles de la triangulation originale, il suffit de tester si sa nouvelle position n'est pas à l'intérieur du cercle circonscrit aux faces miroirs de ses faces adjacents. De façon plus formelle, pour chaque sommet voisin n_i, nous vérifions que la nouvelle position du sommet central est bien à l'extérieur du cercle circonscrit au triangle formé par les sommets n_i, n_{i+1} et m_i. La figure 4.6(a) montre un exemple de site pour lequel le sommet central n'est dans aucun des cercles précédemment définis. Dans cette figure, les cercles colorés sont ceux que l'on doit vérifier.

Pour garantir que le déplacement du sommet central du site ne fait pas apparaître de nouveau triangle, nous proposons un test plus fort que celui nécessaire. Nous vérifions que, si le sommet central du site est à l'intérieur d'un des cercles circonscrits à trois de ces voisins consécutifs n_i, n_{i+1}, n_{i+2}, alors, après modification, il doit toujours se trouver à l'intérieur de ce cercle. S'il en sort, alors une nouvelle face a pu apparaître et la triangulation de Delaunay a pu être modifiée. Même si cette condition n'est pas nécessaire, elle est suffisante pour montrer qu'aucune nouvelle face n'a pu apparaître suite au déplacement. Les cercles circonscrits aux voisins consécutifs



(a) Exemple de cercles circonscrits aux aux triangles miroirs des triangles adjacents au sommet central.



(b) Exemple de cercles circonscrits aux triplets consécutifs de sommets voisins du sommet central.

Le déplacement ne perturbe pas la triangulation de Delaunay, tant que le sommet central du site respecte les deux conditions suivantes : il ne doit pas entrer dans un des cercles présentés dans la figure 4.6(a) et il ne doit pas sortir des cercles illustrés par la figure 4.6(b).

FIG. 4.6 – Exemple des cercles circonscrits intervenant lors du déplacement du sommet central d'un site.

sont illustrés par la figure 4.6(b). Dans cet exemple, la qualité du site n'est pas conservée si le déplacement du sommet central fait sortir celui-ci de l'un de ces cercles.

Définition 4.3.2 *Soit un site* $(c, (n_1, \dots, n_N), (m_1, \dots, m_N), E_c)$ *dont la nouvelle position du sommet central est notée* c'*. La qualité du site est préservée lorsque :*

- *la distance entre* c *et* c' *est inférieure à la perte de précision autorisée ;*
- c' *est bien à l'intérieur du polygone formé par les sommets* n_i *;*
- *pour chaque sommet* n_i *du site,* c' *est à l'extérieur du cercle circonscrit au triangle formé par* n_i*,* n_{i+1} *et* m_i *;*
- *pour chaque sommet* n_i *du site, dans le cas où* c *est à l'intérieur du cercle circonscrit au triangle formé par* n_i*,* n_{i+1} *et* n_{i+2}*,* c' *est aussi à l'intérieur de ce cercle.*

Remarque 2 *Le nombre de tests pour vérifier si une nouvelle face apparaît ou si une face disparaît après modification de la position du sommet central est donc proportionnel au nombre de sommets voisins du sommet central.*

Remarque 3 *Dans la triangulation de Delaunay, le nombre de voisins d'un sommet peut être majoré par une constante ne dépendant pas de la taille du document. On peut donc considérer que l'on peut vérifier la préservation de qualité d'un site en temps constant.*

4.3.3 L'algorithme de tatouage

Le calcul de la triangulation de Delaunay est une étape préliminaire au tatouage. Une fois cette étape achevée, nous nous servons à la fois du document original et de sa triangulation pour énumérer l'ensemble des sites du document. Chaque site extrait est ensuite traité individuellement et localement, sans référence au document global. Il est ensuite réintroduit dans le document original lorsque les modifications sur le site ne dégradent pas la qualité du document. Dans le cas contraire, on n'applique aucune modification dans le document pour ce site.

Seules les fonctions qui permettent d'extraire un site du document et de le réintroduire interagissent avec le document. Le traitement séquentiel des sites est jalonné de plusieurs étapes. Pour chaque site, ces étapes sont les suivantes :

1. la sélection ou non du site en fonction d'une clé et de son codage ;
2. la modification du site si celui-ci est sélectionné, sinon on passe au site suivant ;
3. un test pour déterminer si la modification préserve la qualité du site. Si la qualité du site n'est pas préservée par la modification, alors celle-ci est abandonnée. Sinon on répercute la modification du site dans le document.

Le biais statistique est introduit dans l'ensemble de sites sélectionnés. La composition de cet ensemble dépend de la clé secrète. Ainsi, pour deux clés différentes, l'ensemble de sites choisis est différent. Par conséquent, les documents tatoués sont différents. Comme nous utilisons la même clé pour le tatouage et la détection, ce dernier peut reproduire le sous-ensemble de sites

et retrouver le biais statistique introduit lors du tatouage. Sans connaître la clé, il semble difficile de reproduire la sélection et donc de laver la marque.

Nous répercutons la modification des sites dans le document uniquement dans le cas où la qualité du site est préservée par la modification. Cela signifie qu'une certaine proportion des modifications n'est pas appliquée dans le document. Afin d'obtenir le biais statistique dans le document, l'algorithme de tatouage est conçu de sorte que la proportion de modifications non-appliquées dans le document ne soit pas trop importante.

La figure 4.7 illustre les cinq étapes qui composent le traitement d'un site :

1. extraction du site ;
2. sélection du site ;
3. modification du site ;
4. test de préservation de la qualité du site ;
5. application des modifications du site dans le document.

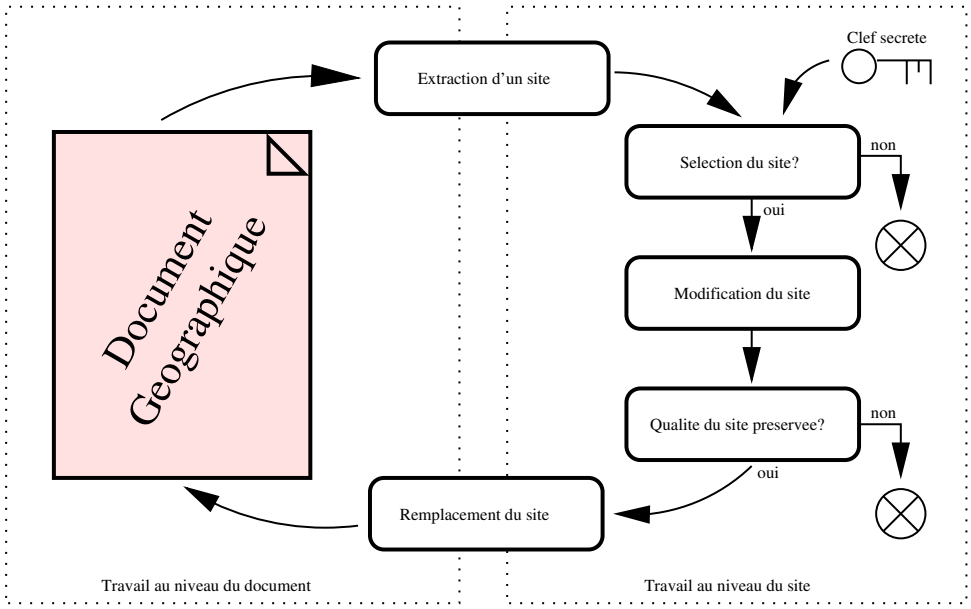


FIG. 4.7 – Schéma général de tatouage.

4.3.4 L'algorithme de détection

L'algorithme de détection est très similaire à l'algorithme de tatouage. Il utilise le document et sa triangulation de Delaunay associée afin d'énumérer les sites présents dans le document. Afin d'harmoniser la définition des algorithmes de tatouage et de détection, nous supposerons que l'algorithme de détection va traiter les sites du document séquentiellement.

L'algorithme de détection reproduit donc le même partitionnement de sites que lors du tatouage. Pour la détection, nous utilisons le même algorithme d'extraction des sites, la même fonction de codage et la même clé secrète.

La détection du biais statistique sur le sous-ensemble de sites sélectionnés est effectuée par dénombrement. On compte le nombre de sites sélectionnés et parmi ceux-ci le nombre de ceux qui satisfont Φ. Sachant que nous avons un modèle de la distribution suivie par la propriété, nous mesurons l'écart entre la distribution modélisée et celle observée. En utilisant la borne de Chernoff[Chernoff, 1952] [Alon et Spencer, 2000], nous pouvons déterminer une majoration de la probabilité d'observer cet écart. La borne obtenue peut être comparée à un seuil déterminé pour répondre, par oui ou non, à la question : « Le document est-il tatoué avec une clé donnée ? » .

La figure 4.8 montre le déroulement de l'algorithme de détection. En la comparant avec la figure 4.7, on voit que les deux premières étapes de l'algorithme de détection : l'extraction des sites et leur sélection, sont les mêmes que pour l'algorithme de tatouage. Lorsque le test a été effectué sur tous les sites, les compteurs servent à mesurer la présence d'un biais statistique.

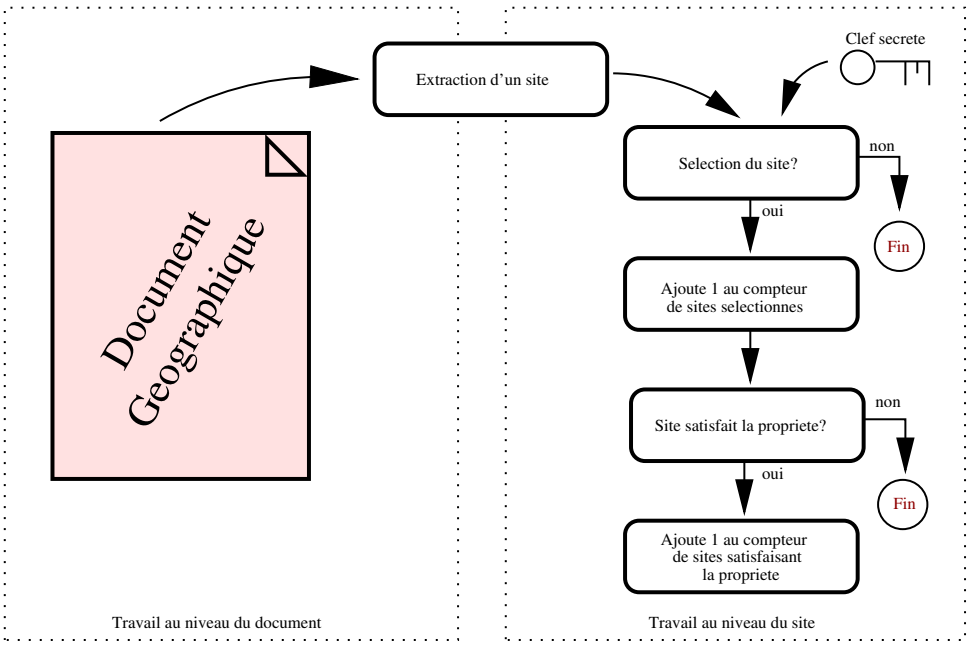


FIG. 4.8 – Schéma général de l'algorithme de détection.

4.4 Détails des étapes de l'algorithme

Dans cette section, nous étudions précisement les différentes étapes des algorithmes de tatouage et de détection. Nous commençons par étudier les étapes d'extraction puis de sélection des sites qui sont communes aux deux algorithmes. Nous détaillons notamment la fonction de codage des sites ainsi que la propriété Φ choisie. Et, en dernier lieu, nous détaillons les algorithmes

de tatouage et de détection.

4.4.1 Extraction des sites

La première étape des algorithmes de tatouage et de détection consiste à extraire les sites d'un document. L'extraction des sites du document est effectuée séquentiellement. Ainsi, nous appliquons la fonction d'extraction sur chaque sommet du graphe pour traiter une seule fois chaque site du document.

Pour obtenir une notion de localité au sein du document, nous calculons une triangulation de Delaunay à partir du nuage des points du document original. L'exemple de la figure 4.9 montre un exemple de cette étape préliminaire La figure 4.9(a) présente un document géographique sous la forme d'une carte vectorielle. L'unicité de la triangulation est garantie par l'utilisation de la bibliothèque de calcul géométrique CGAL [Yvinec, 2007] [CGAL, 2010] qui lève les ambiguïtés comme celles des points cocycliques. La figure 4.9(c) montre la triangulation obtenue. La figure 4.9(d) illustre la superposition du graphe original avec la triangulation.

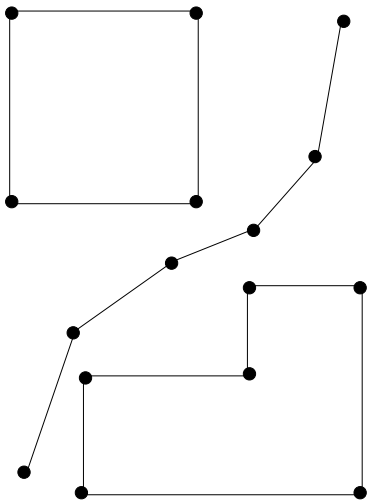
Rappelons que nous extrayons un site pour chaque sommet du document original. En effet, selon la definition 4.3.1, un site est extrait à partir d'un sommet c qui devient le sommet central du site.Pour pouvoir associer un codage à chaque site, nous utilisons les N voisins de c dans la triangulation de Delaunay $n_1, \ldots, n_N$ ainsi que toutes les arêtes entre les sommets $(c, n_1, ..., n_N)$. L'ensemble de ces arêtes est noté E_c. Le site comprend également les sommets miroirs $(m_1, ..., m_N)$ de c nécessaires pour vérifier que la qualité du site est préservée. Les voisins sont triés dans l'ordre trigonométrique, et le premier voisin est choisi arbitrairement. Les sommets m_1 à m_N sont numérotés de façon à ce que le triangle (n_i, m_i, n_{i+1}) appartienne à la triangulation de Delaunay. La figure 4.10 illustre la façon dont un site est extrait du document.

4.4.2 Codage des sites

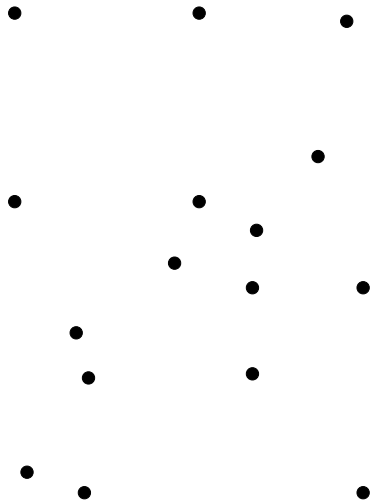
Le codage d'un site repose uniquement sur des critères topologiques, la position précise des sommets n'intervient donc pas dans le codage. De ce fait, le codage sera robuste au déplacement des sommets ou à l'ajout d'un léger bruit tant que la topologie est préservée. Par conséquent, pour une clé fixée, tant que la topologie de chaque site est préservée, on peut reconstituer le partitionnement.

Pour définir le codage d'un site, nous associons à chaque site une matrice binaire $\mathbf{M}$ qui représente la connexité entre les sommets $(c, n_1, ..., n_N)$ où N représente le nombre de voisins de c. La matrice $\mathbf{M}$ est composée de N lignes et de N colonnes. La i-ème ligne représente la connexion de n_i, le i-ème voisin de c avec c ainsi que les connexions entre n_i et les sommets $\{n_1, ..., n_N\}$. Les coefficients de la matrice sont déterminés par les règles suivantes :

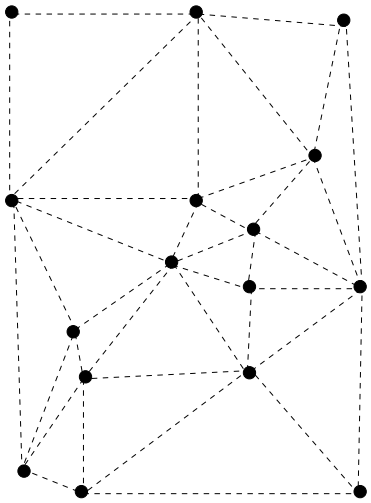
- $\mathbf{M}_{i,1} = 1$ avec $1 \leq i \leq N$, lorsqu'il existe une arête entre n_i et c dans E_c, sinon $\mathbf{M}_{i,1} = 0$;
- $\mathbf{M}_{i,j} = 1$ avec $1 \leq i \leq N$ et $2 \leq j \leq N$, lorsqu'il existe une arête entre n_i et $n_{(i+j-1)\ modulo\ N}$ dans E_c, sinon $\mathbf{M}_{i,j} = 0$.



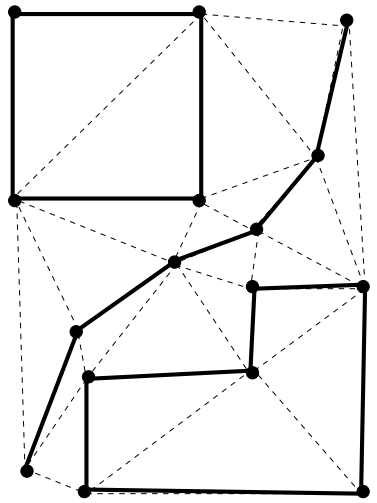
(a) Un exemple de document original.



(b) Le nuage de points extrait du document original.



(c) La triangulation de Delaunay du nuage de points.

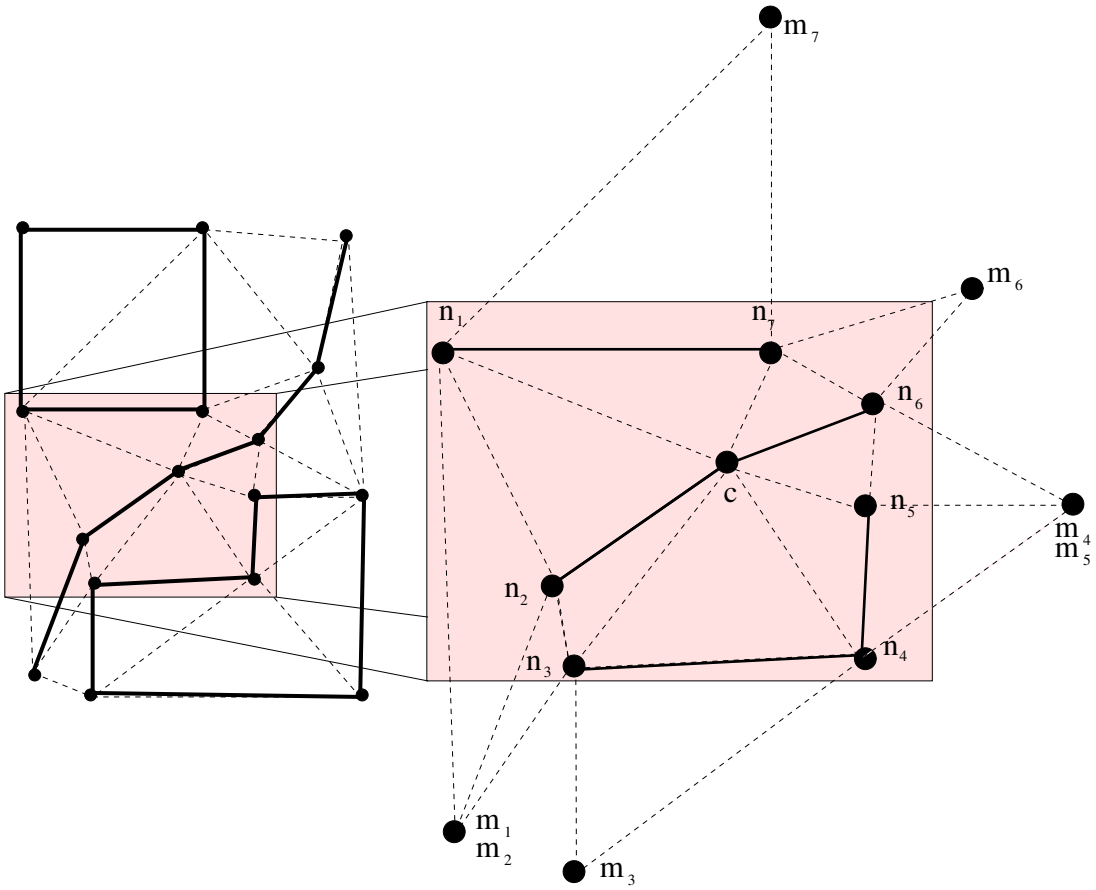


(d) Superposition de la triangulation de Delaunay avec le document original.

FIG. 4.9 – Les étapes préalables à l'extraction des sites.

La matrice calculée dépend du sommet choisi comme premier voisin lors de la numérotation des sommets voisins du sommet central du site. Déterminer un premier voisin pourrait être effectué en utilisant l'orientation (le plus au nord par exemple). Le schéma ne serait alors plus robuste à la rotation par exemple. Pour rendre notre algorithme robuste à la rotation, nous devons rendre le codage indépendant du choix du premier voisin, Pour ce faire, nous calculons la matrice M' qui correspond à la permutation circulaire maximale des lignes de la matrice M

Une rotation circulaire des lignes d'une matrice est obtenue en remplaçant chaque ligne de la matrice par celle qui la précède. Ainsi, la première ligne devient la seconde ligne et la dernière



Ce site comprend un point central noté c, ses voisins dans la triangulation de Delaunay $(n_1, ..., n_7)$ et les arêtes entre ces sommets dans le document original (marquées par des lignes pleines). Enfin, les sommets $(m_1, ..., m_7)$ sont les sommets opposés à c sur les faces miroirs aux faces adjacentes à c.

FIG. 4.10 – Exemple d'extraction d'un site dans le document.

ligne la première.

Pour définir une rotation circulaire maximum, nous définissons une relation d'ordre sur les matrices de mêmes dimensions en prenant l'ordre lexicographique sur les lignes de la matrice concaténées. La rotation circulaire maximum $\mathbf{M}$ est obtenue en choisissant la plus grande des matrices obtenues par rotation circulaire de $\mathbf{M}$.

Le codage du site est obtenu en concaténant les lignes de la matrice $\mathbf{M}'$. Cette matrice est uniquement basé sur la topologie du site et sur les connexions entre les sommets du site. Ainsi, même si le site subit une rotation, ou que ses sommets sont déplacés, tant que sa topologie est conservée, son codage sera conservé.

Prenons un exemple avec le site de la figure 4.10 et calculons la matrice $\mathbf{M}$ associée au site. Pour les connexions associées au sommet n_1, dans la matrice, nous obtenons la ligne suivante $(0, 0, 0, 0, 0, 0, 1)$, car le sommet n_1 est connecté uniquement au sommet n_7. En procédant ainsi,

$$\mathbf{M} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

FIG. 4.11 – Codage du site de la figure 4.10 sous la forme d'une matrice.

$$\mathbf{M}' = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

FIG. 4.12 – Permutation maximale de la matrice 4.11.

nous obtenons la matrice M illustrée par la figure 4.11. Cette matrice permet d'obtenir, par rotation circulaire maximum, la matrice M' (figure 4.12) qui est indépendante du sommet choisi comme premier voisin. En concaténant les lignes de la matrice, on obtient le codage du site :

1000000 0100000 0100001 0000001 1000000 0100000 0000001

4.4.3 Sélection des sites

En fonction de son codage et de la clé secrète, nous décidons si un site doit être :
- forcé à satisfaire la propriété choisie ;
- forcé à ne pas satisfaire la propriété choisie ;
- laissé inchangé.

Cette décision dépend également du paramètre $p \geq 2$ qui sert à régler la proportion de sites du document qui doivent être modifiés.

Nous commençons par partitionner l'ensemble des sites du document en p parties. Chaque site va dans une partie en fonction de son codage et de la clé. Nous introduirons un biais statistique dans la première partie de la partition en augmentant la proportion de sites qui satisfont Φ. Inversement, la proportion de sites de la seconde partie satisfaisant Φ sera diminuée. Les sites appartenant aux autres parties sont laissés inchangés.

La partie à laquelle le site appartient est calculée par la fonction P_p. Cette fonction utilise un hachage modulo p de l'identifiant du site et de la clé. La fonction de hachage que nous avons

utilisée dans la pratique est *MD5sum* [Rivest, 1992].

Définition 4.4.1 (La fonction P_p.) *La fonction P_p, pour une clé k, un paramètre p et un site s dont l'identifiant robuste est représenté par la matrice $\mathbf{M}'$, retourne une valeur comprise entre 0 et $p-1$.*

$$P_p(s,k) = hash(id(s),k)\ modulo\ p$$

Pour un site s donné, pour une clé k fixée et avec la propriété Φ que nous présentons dans la section suivante lorsque :

- $P_p(s,k) = 0$, on forcera le site à ne pas satisfaire la propriété choisie ;
- $P_p(s,k) = 1$, on forcera le site à satisfaire cette propriété ;
- dans les autres cas, le site est laissé inchangé.

Nous verrons expérimentalement que nous obtenons un nombre suffisament grand de sites par partie. De plus, par compensation entre les deux parties modifiées, la proportion de sites satisfaisant la propriété demeure inchangée sur l'ensemble du document. Le chapitre 5 présente une analyse statistique détaillée du partitionnement des sites.

4.4.4 Modification des sites

L'algorithme de tatouage doit modifier les sites sélectionnés pour qu'ils satisfassent ou ne satisfassent pas une certaine propriété que nous notons Φ. Dans cette section, nous définissons cette propriété.

Elle utilise l'aléatoire contenu dans le document. Nous avons choisi une propriété qui soit robuste à un léger déplacement des sommets du site. De plus, à l'instar du codage, la propriété choisie est robuste à de légères transformations géométriques sur le site.

La propriété Φ retenue est la parité de la distance discrète entre le point central du site et le barycentre de ses voisins. Le pas choisi pour discrétiser la distance est la perte de précision autorisée (qui est un paramètre du schéma).

Définition 4.4.2 (Propriété Φ.) *Soit un site $s = (c,(n_1,\dots,n_N),(m_1,\dots,m_N),E_c)$, c est le sommet central du site et $n_1,\dots,n_N$ sont ses voisins. On note δ' la perte de précision autorisée lors du tatouage. On note d la distance entre c et le barycentre des sommets $n_1,\dots,n_N$. Le site s satisfait la propriété lorsque :*

$$\left\lfloor \frac{d}{\delta'} \right\rfloor \equiv 1 \qquad (mod\ 2)$$

Nous avons observé expérimentalement que, dans un document non tatoué, un site satisfait Φ avec une probabilité de μ proche de $0,5$. Les détails de l'expérimentation qui a conduit à obtenir cette estimation sont donnés dans le chapitre 5.

L'exemple présenté par la figure 4.13 illustre la propriété Φ pour un site donné. Les anneaux concentriques sont centrés sur le barycentre des sommets $n_1,\dots,n_N$. La largeur de chaque anneau, blanc ou grisé, est la perte de précision maximale autorisée à l'algorithme de tatouage δ'.

La propriété Φ est satisfaite par un site lorsque le sommet central du site se trouve dans un anneau blanc.

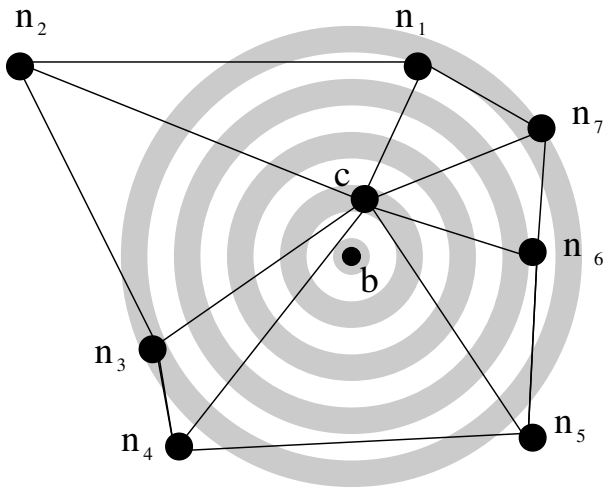


FIG. 4.13 – Visualisation de la propriété Φ, b représente le barycentre des sommets n_1, ...n_7. Si c est dans un anneau blanc, Φ est satisfaite.

Comme nous l'avons vu l'algorithme de tatouage va forcer les sites de la première partie de la partition à satisfaire Φ. Si le site ne satisfait pas déjà cette propriété, il suffit de déplacer le sommet central du site vers le barycentre des voisins d'une distance de δ'. De façon similaire, pour forcer un site à ne pas satisfaire Φ, nous éloignons v du barycentre de ses voisins d'une distance de δ'. On déplace ainsi le site d'un anneau à l'autre en suivant la droite formée par c et le barycentre de ses voisins. Les algorithmes 5 et 6 précisent comment forcer un site à satisfaire ou à ne pas satisfaire Φ.

Algorithm 5: Algorithme pour forcer un site à satisfaire Φ

Input: $s = (c, (n_1, \dots, n_N), (m_1, \dots, m_N), E_c)$: le site qui doit satisfaire Φ
Input: δ' : la perte de précision autorisée
Output: s' : une copie du site s dont le sommet central est déplacé de façon à satisfaire Φ
begin
  $b \leftarrow$ Barycentre des$(n_1, \dots, n_N)$;
  $d \leftarrow$ Distance entre c et b ;
  $v \leftarrow \lfloor \frac{d}{\delta'} \rfloor \bmod 2$;
  if $v = 1$ **then**
    return s
  $c' \leftarrow$ Bouger c vers b d'une distance de δ' ;
  return $(c', (n_1, ..., n_N), (m_1, ..., m_N), E_c)$
end

Algorithm 6: Algorithme pour forcer un site à ne pas satisfaire Φ

Input: $s = (c, (n_1, ..., n_N), (m_1, ..., m_N), E_c)$: le site qui ne doit pas satisfaire Φ
Input: δ' : la perte de précision autorisée
Output: s' : une copie du site s dont le sommet central est déplacé de façon à satisfaire Φ
begin
  $b \leftarrow$ Barycentre des$(n_1, ..., n_N)$;
  $d \leftarrow$ Distance entre c et b ;
  $v \leftarrow \lfloor \frac{d}{\delta'} \rfloor \bmod 2$;
  if $v = 0$ **then**
    return s
  $c' \leftarrow$ Éloigner c de b d'une distance de δ' ;
  return $(c', (n_1, ..., n_N), (m_1, ..., m_N), E_c)$
end

4.4.5 Test de préservation de la qualité des sites

Notre but est de préserver la qualité du document initial lors de la phase de tatouage. On ne souhaite pas appliquer au document une modification qui ne préserve pas sa qualité. Avant d'appliquer la modification du site au niveau du document, nous devons donc nous assurer que celle-ci n'a pas modifié la qualité du site. Si c'était le cas, on risquerait alors de modifier la qualité du document.

Ce test est effectué de façon locale au site. Il s'agit de comparer le site original et le site modifié. Cet algorithme est basé sur la notion de préservation de qualité de site détaillée dans la section 4.3.2. Il utilise le site original, sa version modifiée ainsi que la perte de précision autorisée afin de répondre à la question : « Les deux sites passés en paramètre ont-ils la même qualité sachant que la perte de précision autorisée est δ' ? ».

Nous avons vu précédemment que l'algorithme de modification ne change que la position du sommet central du site, nous avons donc à comparer deux sites qui sont identiques à l'exception de la position de leur sommet central. L'algorithme 7 donne les étapes nécessaires pour tester si la qualité d'un site est préservée après déplacement de son sommet central. Chaque test pour vérifier si un site est à l'intérieur du cercle circonscrit à trois sommets est réalisé à l'aide d'un prédicat de géométrie algorithmique. Ce prédicat, retourne pour un triangle et un point à tester si le point est à l'intérieur, sur la frontière ou à l'extérieur du cercle circonscrit au triangle.

4.4.6 Réintroduction des sites modifiés dans le document

En utilisant l'algorithme de test de préservation de la qualité du site, nous filtrerons les modifications qui risquent de ne pas préserver la qualité du document. Ainsi, nous garantissons que nous réintroduisons, au niveau du document, uniquement les sites dont la qualité est préservée après modification. Pour effectuer la réintroduction du site dans le document, il nous suffit de remplacer la position du sommet du document qui a servi de sommet central au site par la

Algorithm 7: Algorithme de vérification de la qualité d'un site

Input: $s = (c, n_1, \dots, n_N, m_1, \dots, m_N, E_c)$: le site original
Input: $s' = (c', n_1, \dots, n_N, m_1, \dots, m_N, E_c)$: le site modifié
Input: δ' : la perte de précision autorisée
Output: vrai : si s et s' ont la même qualité, faux sinon

```
begin
    if distance(c, c') > δ' then return faux ;
    if c' n'est pas dans le polygone formé par n_1, ..., n_N then return faux ;
    for i de 1 à N do
        j ← (i + 1) mod N ;
        k ← (i + 2) mod N ;
        if c' est dans le cercle circonscrit à (n_i, n_j, m_i) then
            return faux;
        if c est dans le cercle circonscrit à (n_i, n_j, n_k) then
            if c' n'est pas dans le cercle circonscrit à (n_i, n_j, n_k) then
                return faux;
    return vrai ;
end
```

position du sommet central du site modifié.

4.4.7 Le schéma de tatouage

La figure 8 donne l'algorithme de tatouage dont chaque étape a été définie précédemment. Nous avons vu que l'algorithme de tatouage introduit un biais statistique dans le document original. L'algorithme de détection a pour but de détecter la présence de ce biais. De la même façon que l'algorithme de tatouage, l'algorithme de détection extrait les sites d'un document et effectue une sélection des sites en utilisant une clé secrète. Pour vérifier qu'un document a bien été tatoué avec une clé donnée, l'algorithme de détection doit utiliser la même clé pour retrouver le même partitionnement que lors du tatouage.

Pour un document non-tatoué, on vérifie expérimentalement qu'un site vérifie Φ en suivant une loi normale de moyenne μ et d'écart type fixé. À titre indicatif, les expériences présentées dans le chapitre suivant donne une moyenne μ proche de 50% pour un écart-type de $2,55$. Pour un document tatoué, la proportion des sites vérifiant Φ doit s'éloigner de μ dans les deux premières parties de la partition. En connaissant n le nombre de sites d'une partie donnée et m le nombre de sites satisfaisant Φ dans chacune de ces parties, nous mesurons l'écart entre la distribution modélisée et celle observée. La borne de Chernoff, nous permet de déterminer une majoration de la probabilité d'observer cet écart. La borne obtenue peut être comparée à un seuil déterminé pour obtenir une réponse à la question : « Le document est-il tatoué avec une clé donnée ? » .

Algorithm 8: Algorithme de tatouage

Input: $d \in \mathcal{D}$: le document à tatouer
Input: $k \in \mathcal{K}$: la clé secrète
Output: $w \in \mathcal{D}$: le document tatoué
begin
  $w \leftarrow$ copie de d ;
  foreach *site* s *de* w **do**
    $j \leftarrow P_p(s, k)$;
    if $j = 0$ **then**
      if s *satisfait* Φ **then**
        $s' \leftarrow$ Modification de s forcée pour ne pas satisfaire Φ ;
        if *qualité préservée entre* s' *et* s **then**
          Réintroduction de s' à la place de s dans w ;
    else if $j = 1$ **then**
      if s *ne satisfait pas* Φ **then**
        $s' \leftarrow$ Modification de s forcée pour satisfaire Φ ;
        if *qualité préservée entre* s' *et* s **then**
          Réintroduction de s' à la place de s dans w ;
  return w ;
end

Borne de Chernoff Soient n essais de Bernoulli indépendants notés $E_1, ..., E_n$. Pour chaque essai, un succès donne 1 et un échec donne 0. 0n a la probabilité de succès $\Pr(E_i = 1) = \mu$ et la probabilité d'échec $\Pr(E_i = 0) = 1 - \mu$. On définit la variable aléatoire E qui donne le nombre de succès $E = E_1 + E_2 + ... + E_n$. L'espérance, c'est-à-dire le nombre moyen de succès vaut $n\mu$. Donc, $|E - n\mu|$ représente la valeur absolue de l'écart de E par rapport à la moyenne. La borne de Chernoff donne une borne de la probabilité que le nombre de succès observés m s'écarte de la moyenne. On l'obtient par la formule suivante :

$$\Pr(|E - n\mu| \geq |m - n\mu|) \leq 2\, e^{-2\, n\, (\frac{m}{n} - \mu)^2}$$

Afin de reconstituer le même ensemble de sites que lors du tatouage, les sites du document sont extraits en utilisant le même algorithme que lors du tatouage. De plus, la sélection des sites est effectuée en utilisant la même fonction de codage des sites et la même clé secrète.

Soient n_0 et n_1 respectivement le nombre de sites de la première et de la seconde partie et m_0 et m_1 respectivement le nombre de sites de la première et de la seconde partie qui vérifient Φ.

Pour un document non-tatoué, on aura $\frac{m_0}{n_0} \simeq 1 - \mu$ et $\frac{m_1}{n_1} \simeq \mu$. En revanche, pour un document tatoué, les proportions $\frac{m_0}{n_0}$ et $\frac{m_1}{n_1}$ vont se rapprocher respectivement de 0 et de 1.

La probabilité qu'un document ne soit pas tatoué est donné avec E_0 et E_1, deux variables aléatoires qui suivent des lois binomiales de paramètres respectifs $(n_0, 1 - \mu)$ et (n_1, μ).

$$\begin{aligned} \Pr(\text{faux-positif}) \leq \Pr\Big(& |E_0 - n_0(1-\mu)| \geq |m_0 - n_0(1-\mu)| \\ \wedge\ & |E_1 - n_1\mu| \geq |m_1 - n_1\mu| \Big) \end{aligned}$$

Si E_0 et E_1 sont indépendantes, on a :

$$\begin{aligned} \Pr(\text{faux-positif}) \leq\ & \Pr\Big(|E_0 - n_0(1-\mu)| \geq |m_0 - n_0(1-\mu)|\Big) \\ \times\ & \Pr\Big(|E_1 - n_1\mu| \geq |m_1 - n_1\mu|\Big) \end{aligned}$$

Ce qui nous permet de majorer la probabilité qu'un document ne soit pas tatoué :

$$\Pr(\text{document non-tatoué}) \leq 4\, e^{-2\, n_0\, (\frac{m_0}{n_0} - (1-\mu))^2}\, e^{-2\, n_1\, (\frac{m_1}{n_1} - \mu)^2}$$

L'algorithme 9 reprend les différentes étapes qui permettent d'effectuer la détection.

En fixant un seuil λ, il est possible de décider si oui ou non un document est tatoué. Le choix de ce seuil est important et dépend du contexte. En effet, plus ce seuil est petit, plus le risque de considérer non-tatoué un document qui a été tatoué est faible. Par contre, cela augmente le risque de détecter un faux positif. c'est-à-dire de considérer comme tatoué un document qui n'était pas tatoué. Si ce test est utilisé pour amener la preuve qu'une personne a volé un document, il est important de conserver un risque de faux-positif très faible. Il est alors préférabe, dans ce contexte, d'avoir une très forte conviction sur le fait que le document est bien tatoué, au risque de manquer certaines détections. Si ce test est effectué pour estimer la propagation d'un

document sur internet, le seuil de risque de faux positif peut éventuellement être relevé pour ne pas manquer la détection de fichiers tatoués. Dans la section expérimentation, nous verrons comment fixer ce seuil.

Algorithm 9: Algorithme de détection

Input: $w \in \mathcal{D}$: le document à vérifier
Input: $k \in \mathcal{K}$: la clé secrète
Input: $\lambda \in [0,1]$: le seuil de détection
Output: $[Vrai, Faux]$: le document est-il tatoué par la clé k.
begin
  $n_0 \leftarrow 0$; $m_0 \leftarrow 0$;
  $n_1 \leftarrow 0$; $m_1 \leftarrow 0$;
  foreach *site* s *de* w **do**
    $j \leftarrow P_p$(s,k) ;
    if $j = 0$ **then**
      $n_0 \leftarrow n_0 + 1$;
      if s *ne satisfait pas* Φ **then** $m_0 \leftarrow m_0 + 1$;
    else if $j = 1$ **then**
      $n_1 \leftarrow n_1 + 1$;
      if s *satisfait* Φ **then** $m_1 \leftarrow m_1 + 1$;
  return $4\, e^{-2\, n_0\, (\frac{m_0}{n_0} - (1-\mu))^2}\, e^{-2\, n_1\, (\frac{m_1}{n_1} - \mu)^2} < \lambda$;
end

Conclusion sur le schéma de tatouage

Dans ce chapitre nous avons présenté notre schéma de tatouage pour les données géographiques. Nous avons donné les modifications pour lesquelles le schéma doit être robuste ainsi que les qualités du document que nous souhaitions préserver.

Nous avons vu que l'algorithme de tatouage travaille localement sur les sites du document. Certains sites sont sélectionnés en fonction de leur codage et de la clé secrète. Ils sont ensuite modifiés pour satisfaire ou non une propriété Φ. Nous avons conçu l'algorithme de tatouage pour qu'il préserve certaines qualités du document original, aux niveaux topologique et métrique. Pour préserver ces qualités du document, nous avons introduit la notion de préservation qualité de site qui permet de vérifier si une modification locale va perturber la qualité globale du document. Notre algorithme de détection mesure un biais statistique sur la distribution du nombre de sites satisfaisant Φ au sein d'un ensemble de sites sélectionnés par leur codage et la clé secrète. Lorsque le document subit des transformations, la marque est conservée tant que le biais est suffisamment présent.

La notion de site, pour laquelle on définit une qualité locale, une opération de codage et une propriété Φ, peut être généralisée. La partie III présentera notre généralisation du schéma, qui

s'abstrait du type de document et repose sur la notion de site et de préservation de qualité.

Nous avons implémenté l'algorithme et validé que le schéma permet bien de distinguer les documents tatoués de ceux qui ne le sont pas. Le schéma est efficace, il permet de tatouer et de détecter la marque dans des documents de tailles réelles en une ou deux minutes. Dans le chapitre suivant, nous validerons la robustesse du schéma face aux transformations. Nous validerons ensuite expérimentalement certaines hypothèses posées dans cette partie, comme la distributions de Φ par exemple.

Chapitre 5

Évaluation du schéma

Sommaire

Dans ce chapitre, nous nous intéresserons à évaluer la complexité de calcul du schéma avant de déterminer expérimentalement les seuils planchers, c'est-à-dire la taille du plus grand document pour lequel on obtient un faux-négatif, pour différents paramètres du schéma. Nous déterminerons tout d'abord cette limite quand le document n'a pas subit de transformations. Nous répéterons ensuite l'expérience lorsque le document a été découpé ou retatoué.

5.1 Efficacité du schéma

Nous nous intéressons maintenant à évaluer l'efficacité en temps de calcul des algorithmes de tatouage et de détection.

Pour chacun de ces deux algorithmes, on doit calculer une triangulation de Delaunay en 2 dimensions. Pour un document contenant n sommets, cette opération peut s'effectuer en $O(n\ d)$ où d représente le degré maximal d'un sommet dans la triangulation. On traite ensuite chaque site du document. La complexité des opérations sur un site (calculer son codage, vérifier s'il satisfait Φ, le modifier et vérifier si la qualité de site est préservée) dépend du nombre de voisins du sommet central (et donc de son degré). La compléxité du traitement de tous les sites est donc

aussi $O(n\ d)$. Or, le nombre de voisins du sommet central d peut être majoré par une constante ne dépendant pas de n.

On conclut donc que la complexité des algorithmes de tatouage et de détection est en $O(n)$ avec n le nombre de sommets du document.

Par ailleurs, dans l'algorithme de détection, à chaque itération de la boucle principale, on traite un site indépendamment des autres. On peut donc facilement paralléliser la boucle pour rendre plus rapide la détection de la marque. Cette parallélisation, triviale pour l'algorithme de détection, est beaucoup plus compliquée à effectuer pour l'algorithme de tatouage car la modification d'un site influe sur ses voisins. L'ensemble des voisins d'un site est formé des sites qui ont un sommet en commun avec celui-ci. On doit alors synchroniser les processus de sorte que deux processus ne traitent pas en même temps deux sites qui ont un sommet en commun.

Notons que même sans cette parallélisation et sans optimiser particulièrement l'implémentation du schéma, on peut tatouer en quelques minutes un document géographique vectoriel de grande taille. Pour se faire une idée, nous donnons les temps nécessaire pour tatouer et détecter la marque sur les routes du Calvados (document de source IGN), qui contient 159 800 sommets et 170 748 arêtes. Ces temps incluent la conversion du document en graphe et inversement. Le tatouage et la détection de la marque sont calculés en moins de deux minutes sur un ordinateur dont le processeur est cadencé à 2Ghz.

5.2 Conditions expérimentales

5.2.1 Corpus

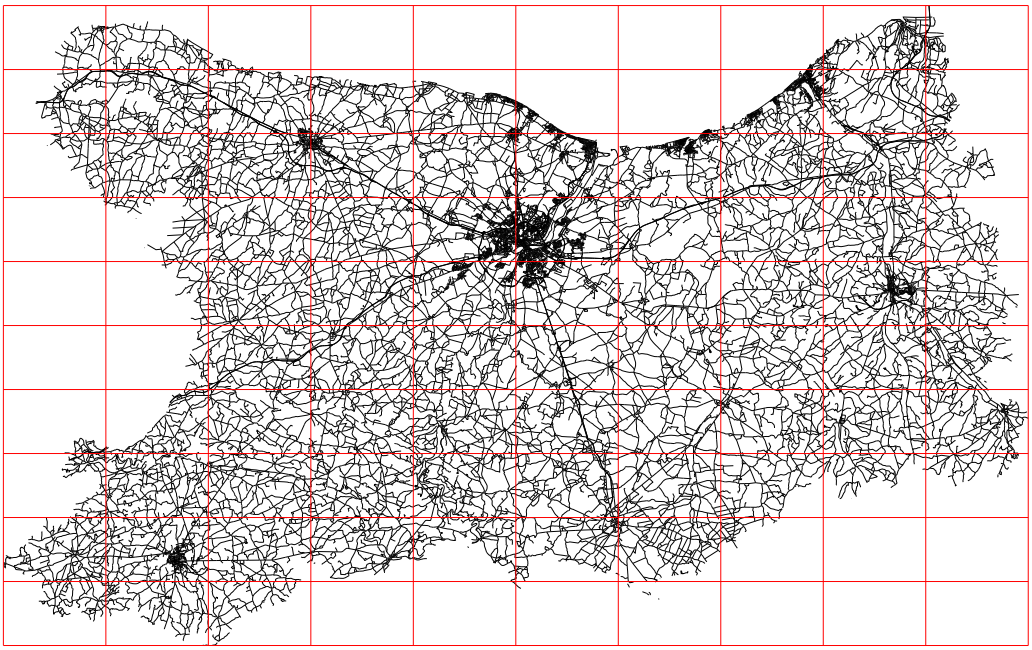
Nous souhaitons étudier notre schéma sur des documents géographiques de natures et de tailles différentes. Pour cela, nous construisons des corpus de documents géographiques à partir de plusieurs documents originaux découpés de différentes façons. Plus on découpe un document, moins nous aurons de sommets dans les documents obtenus.

Le découpage est effectué selon une grille. Chaque case de la grille représente un élément du corpus. La figure 5.1(a) illustre cette opération. Dans cet exemple, nous construisons 100 documents en découpant les données routières du Calvados (source IGN) selon une grille 10×10. Les documents vides sont filtrés et nous ne gardons que 88 documents.

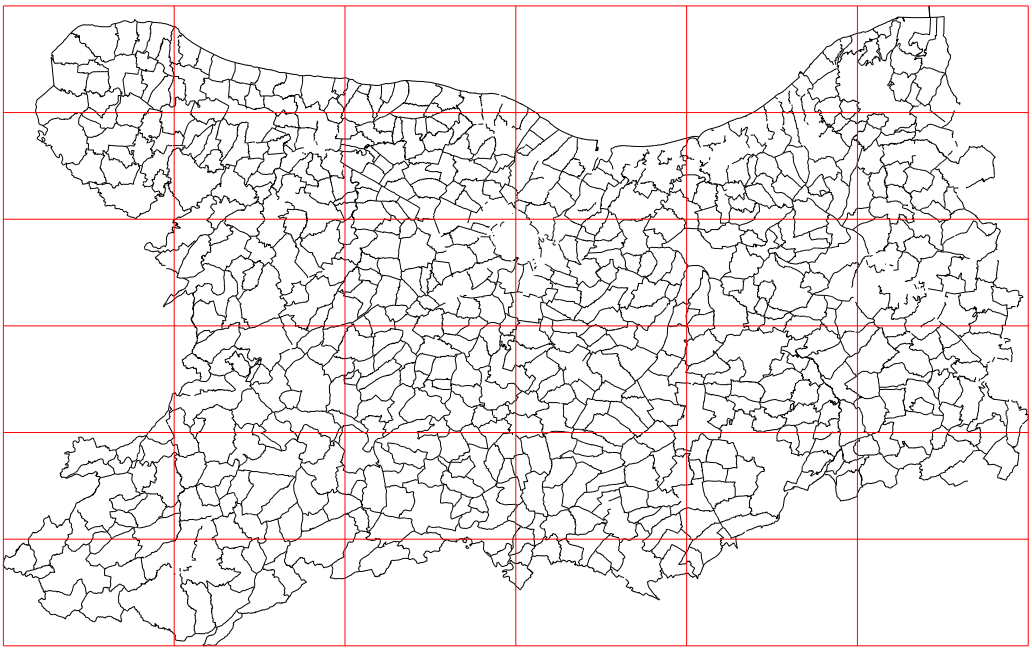
La figure 5.1(b) donne un autre exemple de corpus. Celui-ci est construit en découpant les limites administratives des communes (sources IGN) selon une grille 6×6.

On remarque de grosses différences entre les deux figures. Tout d'abord, les tronçons de routes représenté par la figure 5.1(a) forment un document globalement plus dense que les limites des communes ; il contient plus de sommets et d'arêtes. Par ailleurs, la densité des routes est beaucoup moins homogène que celle des limites des communes. Cela s'explique par le fait que le réseau routier est beaucoup plus dense dans les zones urbaines.

Nous construisons 6 corpus en découpant les limites de communes et les tronçons routiers selon plusieurs grilles : 10×10, 6×6 et 3×3. Les caractéristiques des corpus obtenus sont listés



(a) Découpage des tronçons de routes du Calvados selon une grille de 10×10 (Source IGN).



(b) Découpage des limites des communes du Calvados selon une grille de 6×6 (Source IGN).

FIG. 5.1 – Exemples de découpage des deux documents de départ.

dans le tableau 5.1.

Fichier original	limites des communes			tronçons de routes		
Nombre de sommets	66 703			159 800		
Nombre d'arêtes	66 858			170 748		
Grille de découpage	3×3	6×6	10×10	3×3	6×6	10×10
Nombre de documents non-vides produits	9	35	88	9	35	88
Minimum de sommets	3 436	49	6	5 245	80	5
Nombre moyen de sommets	7 411	1 905	757	17 754	4 564	1 815
Maximum de sommets	10 921	3 208	1 358	41 202	18 664	15 078
Écart type du nombre de sommets	2 141	928	348	9 493	3 666	1 952

TAB. 5.1 – Description des 6 corpus utilisés

5.2.2 Dispositif expérimental

Notre schéma est paramétré par la clé de tatouage, le nombre de parties de la partition et la perte de précision autorisée. Afin d'étudier précisément l'impact de chacun des paramètres sur le schéma, nous avons construit un dispositif expérimental qui exécute automatiquement des expériences en fonction d'une liste de valeurs fixées pour chaque paramètre. Nous exécutons une expérience pour chaque combinaison de valeurs possibles des paramètres sur chaque corpus.

Par exemple, pour vérifier que nous arrivons bien à insérer puis à détecter une marque, nous faisons varier les paramètres suivants :

- la clé de tatouage qui prend 3 valeurs différentes (jacques, jeanmarie et HHH) ;
- le nombre de parties, qui prendra les valeurs 2, 4, 8, 10, 12 ou 16 ;
- la perte de précision autorisée, 1, 3 ou 5 mètres.

Pour cet exemple, la combinaison de tous les paramètres produit 54 expériences, que l'on renouvelle sur chacun des 6 corpus pour obtenir 324 expériences. Les expériences sont exécutées en parallèle, leurs résultats sont stockés. À partir de ces résultats, nous produisons des rapports pour chaque expérience ainsi que des rapports de synthèse. Ces rapports prennent la forme de tableaux ou de graphiques. Dans cette partie, nous présenterons certains rapports d'expériences qui nous semblent significatifs ainsi que les rapports de synthèse.

Afin de déterminer si le document est tatoué ou non, nous utilisons un seuil. Les résultats des expériences donnent la proportion de faux-positifs et de faux-négatifs pour notre corpus de test, pour 5 valeurs de seuil λ différentes 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4} et 10^{-5}.

5.3 Détection de la marque

Nous commençons par vérifier que nous discriminons bien les documents tatoués des originaux.

Le protocole consiste à tatouer chaque document du corpus. Avec la même clé, nous détectons ensuite la présence de la marque sur les documents tatoués obtenus et les originaux.

La figure 5.2 donne la probabilité de présence du biais statistique mesuré sous l'hypothèse que le document testé n'est pas tatoué en fonction de la taille du document exprimée en nombre de sommets. Sur cette figure, chaque document est représenté par un point. Les points violets et blancs représentent respectivement les documents tatoués et ceux qui ne le sont pas. L'abscisse du point est le résultat de la borne pour l'algorithme de détection pour le document (que l'on va comparer au seuil). L'ordonnée du point représente le nombre de sommets du document. Sur ce graphique, plus les points colorés sont à gauche et plus les blancs sont à droite, meilleure est la détection de la marque. Afin de mieux illustrer le résultat de l'expérience, nous avons représenté les documents sur la gauche du graphique quand la borne de Chernoff calculée est inférieure à 10^{-20}.

Ainsi, dans la figure 5.2, nous avons paramétré une perte de précision autorisée de 1 mètre, et une partition des sites en 4 parties. Nous voyons clairement que l'algorithme de détection permet de bien discriminer les deux groupes de documents. La distinction entre les deux groupes s'accroît lorsque le nombre de sommets du document augmente.

Le tableau récapitulatif 5.2 donne le nombre de faux-négatifs et de faux-positifs pour notre corpus et plusieurs valeurs du seuil de discrimination. On voit que pour les documents de 100 sommets, la détection fonctionne parfaitement pour un seuil à partir de 10^{-2}.

Seuil	Documents de plus de :					
	0 sommet (tous)		100 sommets		200 sommets	
	88 documents		84 documents		81 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	5	4	5	0	5	0
10^{-2}	0	4	0	0	0	0
10^{-3}	0	4	0	0	0	0
10^{-4}	0	4	0	0	0	0
10^{-5}	0	4	0	0	0	0

TAB. 5.2 – Résultat de la détection sur le corpus de test (88 documents).

Afin de synthétiser les résultats de toutes nos expériences, nous avons construit le tableau 5.6. Celui-ci donne le nombre de sommets du document le plus grand pour lequel la détection a échouée. Plus ce nombre est petit, plus le détection de la marque fonctionne sur de petits documents. Nous donnons les résultats pour deux documents différents en faisant varier la perte de précision autorisée et le nombre de parties. Pour chaque case du tableau, nous réunissons les résultats des expériences obtenus en tatouant avec 3 clés différentes.

Cette figure montre que plus on choisit un seuil de détection faible, plus le risque de faux-négatifs augmente pour de petits documents. De même, on constate que plus on augmente la

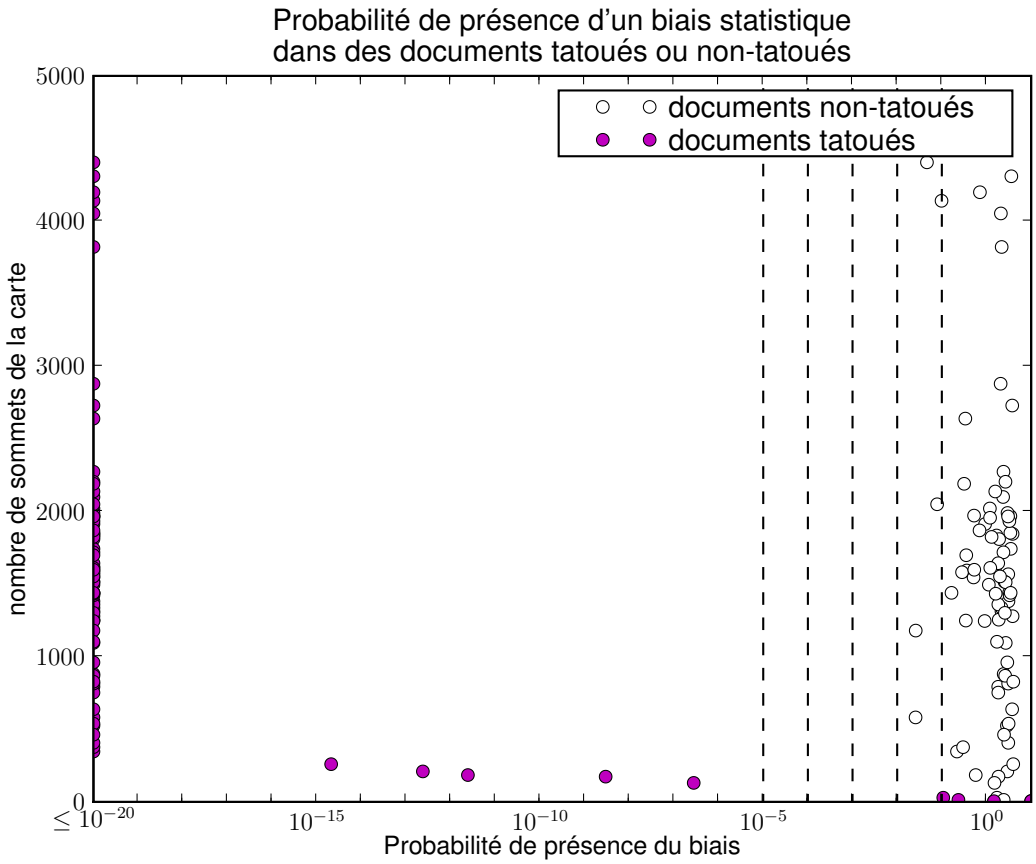


FIG. 5.2 – Résultat de la détection sur les tronçons de routes découpés selon une grille 10×10. Chaque document est tatoué avec un partitionnement en 4 parties et avec une perte de précision autorisée de 1 mètre.

perte de précision autorisée, plus la marque est difficile à détecter. Cela s'explique par le fait que lorsqu'on augmente la perte de précision autorisée, on a plus de chances de modifier la qualité du site et donc de devoir annuler la modification. Par conséquent, moins de sites sont modifiés, ce qui donne une marque moins présente dans le document. Enfin, on constate que moins on a de parties, plus la marque peut être détectée dans de petits documents. En effet, moins la partition contient de parties, plus le nombre de sites concernés par le tatouage est important.

Dans le tableau, on remarque que le partitionnement en 12 parties semble moins bien fonctionner que celui en 16 parties. L'origine de ce problème semble venir du fait que la répartition des sites n'est pas uniforme dans les parties. En fonction de la clé utilisée et du nombre de parties de la partition, on peut avoir plus ou moins de sites dans les deux premières parties. En d'autres termes, on peut avoir plus ou moins de sites impliqués dans le tatouage. Pour se convaincre de cette hypothèse, le tableau 5.7 montre la même expérience en séparant chacune des 3 clés (la perte de précision autorisée est fixée à 1 mètre). On voit que la seconde clé donne des résultats particulièrement mauvais pour un partitionnement en 12 parties. Le partitionnement des sites en 12 parties avec cette clé donne très peu de sites dans les deux premières parties. Le nombre de sites concernés par le tatouage est donc plus faible. C'est ce mauvais résultat que l'on retrouve dans le tableau de synthèse 5.7.

Nous pouvons aussi expliquer que la détection reste correcte avec 16 parties par le fait que moins l'on a de sites impliqués dans le tatouage, moins on risque d'avoir de couples de sites voisins impliqués dans le marquage. Nous appelons sites voisins, deux sites dont le sommet central de l'un se trouve parmi les voisins de l'autre. Dans ce cas, la modification d'un des sites peut changer la satisfaction de Φ de ses voisins. Donc, en modifiant moins de sites, on réduit les chances de changer la satisfaction de Φ pour d'autres sites.

Pour conclure, cette expérience nous apprend que le schéma de tatouage fonctionne bien, même pour des documents très petits. Avec un partitionnement en 4 parties et une perte de précision autorisée de 1 mètre, il est possible d'insérer puis de détecter une marque dans un document issu des tronçons routiers du Calvados qui contient moins de 100 sommets. Nous avons aussi montré que moins on a de parties et plus la perte de précision autorisée est faible, plus la marque peut être détectée dans de petits documents. Cette expérience a permis de constater que le choix de la clé à une forte influence sur la détection de la marque. Nous verrons dans la section 6 que cela vient du fait que les sites ne sont pas distribués uniformément entre les parties.

5.4 Étude de robustesse aux transformations légitimes

5.4.1 Robustesse aux transformations géométriques et au changement de l'ordre des objets

Dans cette section, nous étudions la robustesse du schéma contre certaines des transformations présentées dans la section 4.1.4. Nous allons montrer que, par définition, notre schéma est

robuste à la réorganisation des données, la translation et la rotation. Dans la partie expérimentation, nous étudierons la robustesse du schéma aux autres transformations considérées : découpage et retatouage.

L'algorithme de détection est une chaîne de traitements. Pour influer sur le résultat de l'algorithme de détection, il faut qu'une transformation change le résultat d'au moins une des étapes de la détection. L'extraction des sites est basée sur la topologie et la triangulation de Delaunay du document. Le codage utilise l'aspect topologique du site. Tant qu'une transformation n'altère pas la topologie et la triangulation du document, le résultat de l'extraction et du codage des sites sera préservé. De façon similaire, si la transformation ne modifie pas du tout les critères métriques locaux aux sites, le site continuera de vérifier ou non Φ.

Pour autant, certaines transformations peuvent modifier les critères topologiques ou métriques du document sans pour autant nuire au résultat final de l'algorithme de détection. En effet, l'algorithme de détection vérifie si une proportion anormale de sites vérifient Φ. Si la proportion est amoindrie, la détection sera affectée, cependant, tant que ce biais reste suffisant, on pourra encore détecter la marque. Le découpage et le retatouage entrent dans cette classe de transformations, la robustesse du schéma face à ces transformations sera montrée expérimentalement.

Robustesse à la translation et à la rotation

La translation et la rotation du document ne modifient pas la topologie du document. Par conséquent, les étapes d'extraction des sites et de codage des sites ne sont pas perturbés par ces transformations. De plus, comme les critères métriques locaux à chaque site sont préservés par ces transformations, le calcul de la valeur associée à chaque site est la même avant et après une translation ou une rotation.

Nous montrons donc qu'aucun des résultats des étapes de la détection n'est altéré ni par rotation ni par translation du document. Notre schéma résiste donc à ces deux transformations.

Robustesse au changement de l'ordre des objets

La première étape de l'algorithme consiste à extraire tous les sommets du document. Lors de cette étape, nous construisons une triangulation de Delaunay à partir des sommets. La triangulation obtenue ne dépend pas de l'ordre dans lequel les sommets sont donnés. De plus, l'ordre dans lequel apparaissent les sommets n'a pas d'importance pour le reste de l'algorithme car il utilise la notion de site. Par conséquent, appliquer une transformation qui consiste à changer l'ordre des objets dans le document n'aura pas d'effet sur l'algorithme de détection. Le schéma est donc robuste à ce type de transformation.

5.4.2 Robustesse au découpage

Nous venons de montrer que notre schéma permet d'insérer une marque dans un document puis de la détecter. Nous nous intéressons maintenant à étudier sa robustesse face au découpage. Pour mémoire, l'opération de découpage consiste à supprimer les points du document qui ne

font pas partie d'un rectangle choisi. En pratique, cette transformation est tout à fait naturelle, elle est utilisée par exemple pour isoler une ville dans un document géographique représentant une région. Ce genre de transformation implique une perte d'information qui va obligatoirement influer sur l'algorithme de détection. Pour tester la résistance de notre schéma au découpage du document, nous avons procédé à des expérimentations.

Le protocole expérimental pour cette expérience consiste à découper un document selon une grille de 4×4. Chacun des 16 documents obtenus est tatoué. On découpe ensuite 20 fois chaque document selon des rectangles choisis aléatoirement.

La figure5.3 illustre le résultat d'une de ces expériences. Pour celle-ci nous avons utilisé le document des tronçons de routes du Calvados, fixé la taille de partition à 4 et la perte de précision à 1 mètre. Le découpage nous fait manquer quelques détections pour de petits documents, de 100 à 400 sommets.

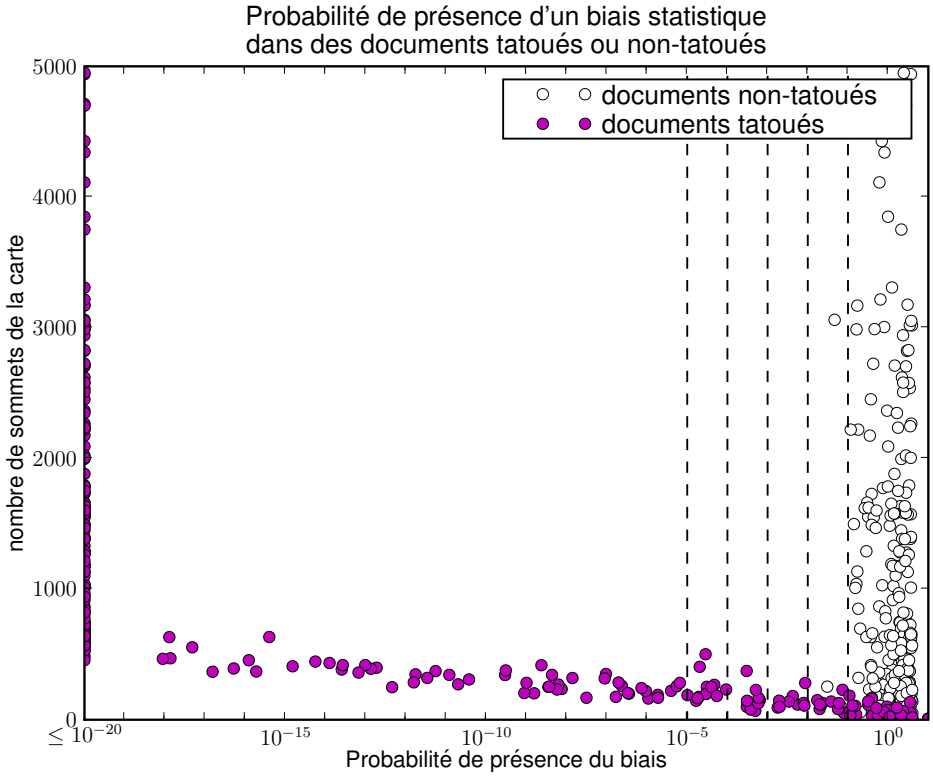


FIG. 5.3 – Résultat du test de découpage de la carte sur les tronçons de routes du Calvados avec un partitionnement en 4 parties et une perte de précision autorisée de 1 mètre.

Le tableau 5.3 donne une synthèse du graphique. Sur ce tableau, on voit clairement que le nombre de détections manquées (faux-négatifs) est plus importante pour les documents de moins de 200 sommets. La discrimination devient correcte pour les documents qui contiennent au moins 400 sommets.

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	289 documents		209 documents		164 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	3	42	2	0	1	0
10^{-2}	0	53	0	1	0	0
10^{-3}	0	62	0	2	0	0
10^{-4}	0	72	0	4	0	0
10^{-5}	0	84	0	10	0	2

TAB. 5.3 – Résultat de la détection après découpage sur le corpus de test.

Les tableaux 5.8 donnent la taille du plus grand document tatoué pour lequel la détection est manquée. Les valeurs évoluent de la même façon que pour le tableau 5.6. Cependant, lorsque l'on compare les valeurs des deux tableaux, on constate que celles du tableau 5.8 sont globalement plus élevées.

En effet, lorsque l'on s'intéresse au découpage du document, on voit que la plupart des sites qui se trouvent dans la zone découpée sont inchangés. Les seuls sites qui subissent des modifications sont ceux qui se trouvent sur le pourtour de la zone découpée. Ces sites sont facilement identifiables car leur sommet central se trouve sur l'enveloppe convexe de la triangulation. Dans la section 7.1, nous proposerons la même expérience en filtrant les sites dont les sommets centraux sont sur l'enveloppe convexe.

5.4.3 Robustesse au retatouage

Nous évaluons la résistance du schéma contre le *retatouage*. Rappelons que cette transformation consiste à prendre un document tatoué avec une clé et à le tatouer à nouveau avec une autre clé. Nous allons évaluer l'impact de ce second tatouage sur la détection de la première marque.

Il est intéressant d'étudier cette transformation car elle peut être utilisée par un utilisateur malveillant pour s'approprier le document. On sait que l'algorithme de tatouage est public, il est donc facile pour n'importe qui d'introduire sa propre marque dans le document en utilisant l'algorithme de tatouage. La marque originale doit être encore décelable après cette transformation.

Avec notre schéma, cette transformation va produire un document qui est tatoué par les deux clés. Il existera donc deux versions publiées du document, une tatouée par le propriétaire légitime et une autre tatouée à la fois par le propriétaire original et par l'utilisateur malveillant. Dans ce cas, le propriétaire légitime peut faire valoir ses droits en montrant qu'il n'existe pas de version du document qui ne soit pas tatouée avec sa clé.

La figure 5.4 illustre le résultat de l'expérience sur les tronçons de routes du Calvados pour

une partition en 4 parties, une perte de précision autorisée de 1 mètre et un couple de clés fixé.

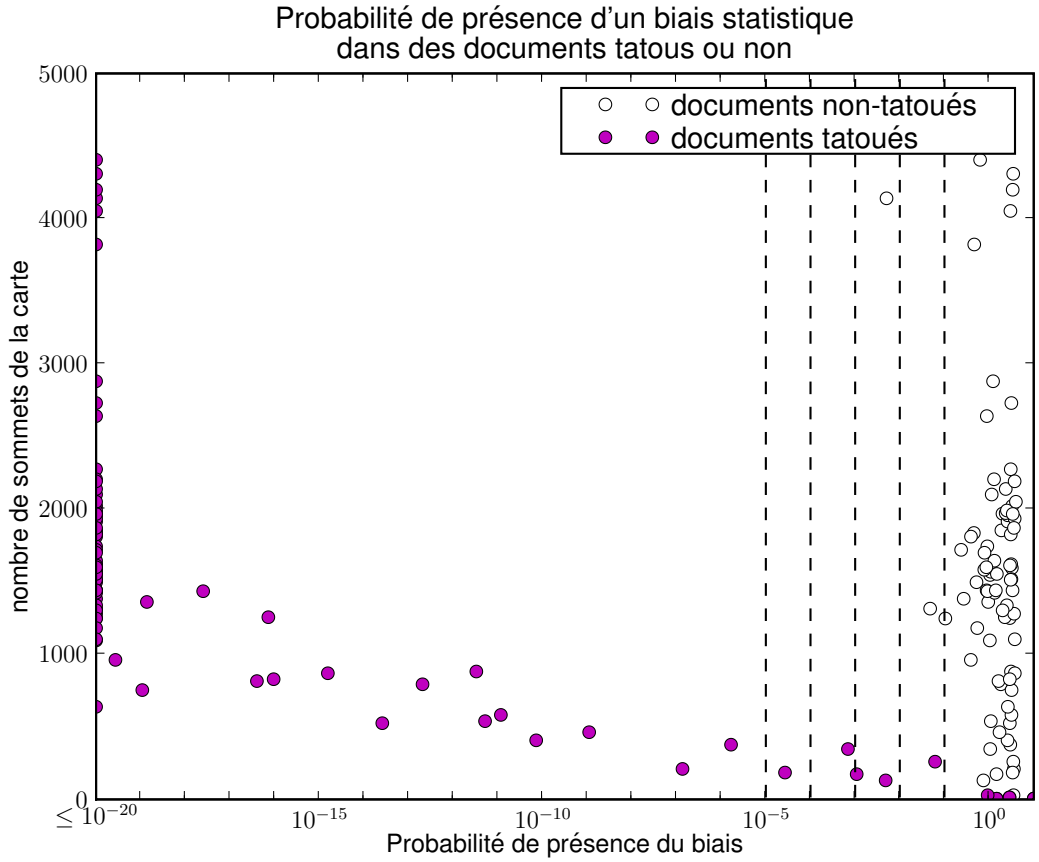


FIG. 5.4 – Résultat du test de détection après retatouage sur les tronçons de routes du Calvados (partition en 4 parties, perte de précision autorisée de 1 mètre).

Le tableau 5.4 donne le nombre de faux positifs et de faux négatifs pour les documents de plus de 100 et de plus de 200 sommets. Ce tableau montre que, pour l'exemple, on arrive bien à discriminer les documents tatoués de ceux qui ne le sont pas à partir de 200 sommets avec un seuil de détection à 10^{-2}.

La figure 5.5 donne le résultat d'une seconde expérience avec les mêmes paramètres, la différence réside dans le fait que l'on effectue l'opération de retatouage avec une autre clé. Cette expérience nous donne des résultats différents. Lorsqu'on compare les graphiques des deux expériences de retatouage, on constate que la seconde donne de moins bon résultats. Cela s'explique par le fait que les partitionnement obtenus par deux clés différentes peuvent être plus ou moins corrélées. Ce phénomène et ses conséquences sur le schéma seront étudiés plus en détails dans la section 6.

Le tableau 5.9 donne la taille du plus grand document (en nombre de sommets) pour lequel l'expérience a échoué. Cela signifie que pour tous les documents contenant plus de sommets,

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	88 documents		84 documents		81 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	3	4	3	0	3	0
10^{-2}	2	5	2	1	2	0
10^{-3}	1	7	1	1	1	0
10^{-4}	1	8	1	2	1	0
10^{-5}	0	9	0	2	0	0

TAB. 5.4 – Résultat de la détection après retatouage.

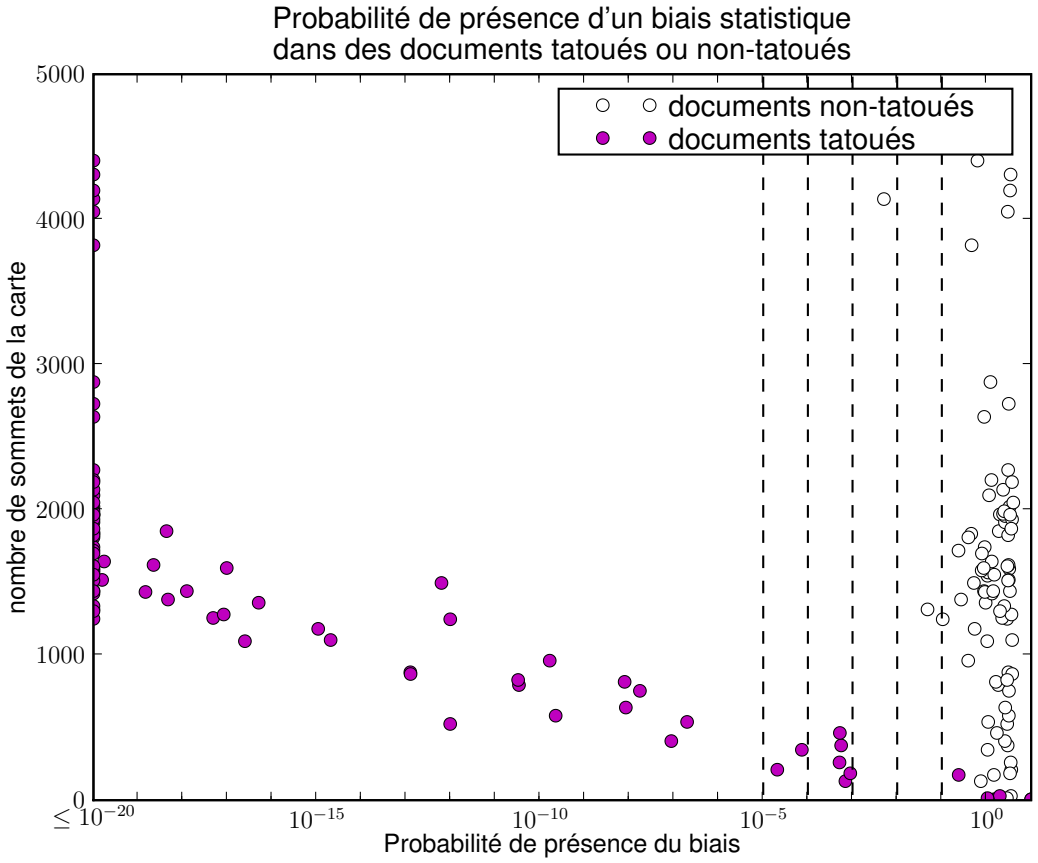


FIG. 5.5 – Second résultat du test de détection après retatouage sur les tronçons de routes du Calvados (partition en 4 parties, perte de précision autorisée de 1 mètre).

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	88 documents		81 documents		77 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	3	5	3	0	3	0
10^{-2}	2	5	2	0	2	0
10^{-3}	1	5	1	0	1	0
10^{-4}	1	10	1	3	1	1
10^{-5}	0	12	0	5	0	1

TAB. 5.5 – Second résultat de la détection sur le corpus de test après retatouage (88 documents).

l'expérience a réussi. Pour chaque paramètre, nous avons renouvelé l'expérience avec 3 couples de clés différentes. Dans ce tableau, on fait varier le nombre de parties de la partition et la perte de précision autorisée. En analysant le tableau, on peut remarquer plusieurs choses. Tout d'abord, le retatouage ne fonctionne pas très bien lorsqu'on utilise un partitionnement en deux parties. En effet, dans ce cas, tous les sites impliqués dans le tatouage avec la première clé sont impliqués dans le tatouage avec la seconde. On a donc toutes les chances de laver la marque. Par ailleurs, on semble constater une amélioration de la détection lorsqu'on tatoue avec une partition des sommets en 16 parties. Nous avons deux explications :

- les partitionnements pour deux clés différentes peuvent être plus ou moins corrélées suivant les clés ;
- moins on modifie de sites, moins on risque de changer la satisfaction de Φ des sites de leurs voisinages.

Dans la partie 6, nous étudierons plus en détails ces deux explications.

Conclusion sur la validation du schéma

Dans ce chapitre, nous avons vu que notre schéma permet bien de discriminer les documents tatoués de ceux qui ne le sont pas, même pour de très petits documents (d'au moins 100 sommets). Nous avons aussi étudié la robustesse du schéma contre le découpage et le retatouage. Les expériences ont montré que la détection de la marque fonctionne bien pour de petits documents, même lorsque le document a subi ces transformations.

Les sites qui se trouvent sur l'enveloppe convexe de la triangulation d'un extrait découpé d'un document tatoué vont perdre leur codage. Ils risquent donc de nuire à la détection de marque. Nous verrons qu'en filtrant ces sites la robustesse du schéma au découpage peut être améliorée lorsque l'échantillon découpé est petit. La section 7.1, présentera des expérience pour valider cette variante.

Nous avons vu que le résultat du retatouage dépend des deux clés utilisées pour tatouer puis retatouer. Comme nous le montrerons dans la section 6.2.2, il existe une corrélation entre les

partitionnements pour deux clés différentes. Dans la section 7.2 nous présentons une variante du schéma afin de rendre indépendants les partitionnements des sites pour deux clés différentes.

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	14	29	29	29	80	80	29	29	80	155	347	155	29	80	131	185	281	311
10^{-2}	29	29	80	80	210	155	29	29	155	174	347	311	29	80	377	185	525	377
10^{-3}	29	80	80	185	210	174	29	29	185	185	347	377	80	185	377	377	881	539
10^{-4}	29	80	155	185	347	185	80	155	185	377	881	377	80	185	377	881	881	793
10^{-5}	29	80	185	210	377	377	80	174	260	377	881	463	80	185	377	881	881	1180

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5						10
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16	12
Seuil de détection																			
10^{-1}	49	49	49	49	174	103	34	49	236	165	202	236	49	103	236	198	278	477	0
10^{-2}	49	49	103	103	199	236	49	174	236	199	278	278	103	103	566	477	768	800	0
10^{-3}	49	103	202	198	278	236	103	174	278	566	278	477	103	236	800	477	998	800	0
10^{-4}	49	103	202	228	278	236	174	174	278	566	564	477	103	236	800	694	998	948	0
10^{-5}	49	174	236	236	353	236	174	174	477	566	1034	708	103	278	800	800	1198	1034	0

TAB. 5.6 – Taille du plus grand document (en nombre de sommets) pour lequel la détection de la marque a échoué (Les corpus issus des mêmes documents sont regroupés et chaque expérience est renouvelée avec 3 clés différentes).

(a) Sur les tronçons de route du Calvados

Clé de tatouage	HHH						jacques						jeanmarie					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	29	80	29	80	174	311	14	29	80	174	281	281	14	29	131	185	347	260
10^{-2}	29	80	281	131	174	377	29	29	281	174	525	281	29	29	377	185	347	377
10^{-3}	29	185	311	174	311	539	80	80	281	377	525	281	29	174	377	347	881	377
10^{-4}	29	185	311	377	311	793	80	174	281	377	539	377	29	174	377	881	881	377
10^{-5}	80	185	311	377	407	1180	80	174	377	377	881	407	80	174	377	881	881	539

(b) Sur les limites administratives des communes du Calvados

Clé de tatouage	HHH						jacques						jeanmarie					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	24	103	236	94	236	236	24	49	236	103	228	236	49	49	236	198	278	477
10^{-2}	103	174	236	165	236	236	49	103	236	278	768	236	103	103	566	477	477	800
10^{-3}	103	236	236	236	477	764	103	174	236	278	998	764	103	236	800	566	684	800
10^{-4}	174	236	442	353	800	903	103	236	236	694	998	764	103	236	800	595	694	948
10^{-5}	174	236	477	353	903	1034	103	278	400	778	1198	1034	103	236	800	800	1027	948

TAB. 5.7 – Taille du plus grand document (en nombre de sommets) pour lequel la détection de la marque a échoué (La perte de précision autorisée est de 1 mètre).

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	281	281	373	343	501	373	229	316	418	501	501	554	280	372	632	444	665	586
10^{-2}	281	281	405	501	501	501	280	502	502	501	820	571	280	502	632	554	1011	1011
10^{-3}	281	316	501	586	571	501	280	502	554	646	1177	1011	373	632	641	867	1103	1011
10^{-4}	281	373	501	632	849	554	502	502	820	666	1177	1011	500	632	1011	867	1553	1381
10^{-5}	281	501	554	632	849	643	502	502	1011	666	1262	1634	500	632	1011	967	4428	1578

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	162	263	294	283	423	398	224	372	423	452	772	452	373	372	969	521	767	969
10^{-2}	162	263	294	403	441	423	224	452	452	490	772	767	373	423	969	969	812	969
10^{-3}	263	403	423	403	631	452	263	452	647	767	814	807	422	526	969	969	1219	1139
10^{-4}	263	403	423	454	631	675	373	452	767	969	969	969	422	647	969	1012	1318	1224
10^{-5}	263	423	643	490	772	675	373	452	767	969	969	969	423	647	969	1012	1359	1350

TAB. 5.8 – Nombre maximum de sommets pour lequel la détection de la marque a échoué après découpage (Les corpus issus du même document original sont fusionnés).

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	13680	174	155	185	347	155	4763	174	260	155	347	260	2872	377	377	377	881	377
10^{-2}	13680	260	260	260	347	347	4763	377	377	311	881	463	2872	377	377	881	881	793
10^{-3}	18357	281	377	260	347	377	9017	377	539	539	881	539	4232	407	793	881	1846	881
10^{-4}	18357	463	463	407	539	463	9017	539	539	539	881	828	4232	793	1311	881	1968	5245
10^{-5}	20013	463	463	638	828	463	9017	869	881	753	1360	1246	4699	881	1311	881	2640	5245

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	3436	202	198	353	353	236	2667	477	477	800	353	477	1692	892	825	903	1027	800
10^{-2}	5620	400	278	859	353	353	2745	906	786	928	948	477	2354	1311	955	1299	1027	1034
10^{-3}	6829	766	477	859	442	477	2966	906	906	1238	1034	948	2966	1311	1142	1692	1232	2442
10^{-4}	10921	906	634	955	726	477	3208	948	948	1311	1086	1074	2966	1311	1299	1692	2083	2442
10^{-5}	10921	1182	825	1238	849	531	6829	1033	1182	1311	1269	1299	3027	1311	1311	2733	2083	2442

TAB. 5.9 – Taille du plus grand document pour lequel la détection de la marque a échoué après retatouage
(Les corpus issus du même document original sont fusionnés, les expériences avec des couples de clés différentes ont été fusionnées).

(a) Sur les tronçons de route du Calvados

Clé de tatouage	('jacques', 'HHH')						('jeanmarie', 'HHH')						('jeanmarie', 'jacques')					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	2100	377	377	377	525	281	1314	377	377	260	753	377	13680	260	377	185	881	377
10^{-2}	2640	377	377	377	525	281	1716	377	377	347	881	793	13680	377	377	881	881	377
10^{-3}	4232	407	377	539	881	377	1716	377	793	881	1846	881	18357	377	753	881	1360	793
10^{-4}	4467	793	407	539	881	407	1853	793	881	881	1968	881	18357	463	1311	881	1835	5245
10^{-5}	4467	793	525	753	1311	638	2050	869	881	881	2640	1311	20013	881	1311	881	1835	5245

(b) Sur les limites administratives des communes du Calvados

Clé de tatouage	('jacques', 'HHH')						('jeanmarie', 'HHH')						('jeanmarie', 'jacques')					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	1955	400	400	859	768	236	2079	892	800	903	1027	800	3436	477	825	566	674	800
10^{-2}	2745	442	400	1027	998	694	2967	1311	955	1299	1027	1034	5620	477	955	800	1027	948
10^{-3}	2745	840	442	1238	1034	1198	3436	1311	955	1692	1142	2442	6829	825	1142	800	1232	1033
10^{-4}	3109	1033	786	1238	1311	1269	3436	1311	1299	1692	2083	2442	10921	1142	1150	947	2083	1299
10^{-5}	3208	1077	1033	2733	1311	1269	3436	1311	1311	2083	2083	2442	10921	1142	1311	948	2083	2442

TAB. 5.10 – Taille du plus grand document pour lequel la détection de la marque a échoué après retatouage
(Les corpus issus du même document original sont fusionnés, la perte de précision autorisée est fixée à 1 mètre).

Chapitre 6

Étude statistique du schéma

Sommaire

Dans ce chapitre, nous nous intéressons aux mécanismes qui permettent au schéma de fonctionner. Nous validerons certaines hypothèses posées dans le chapitre précédent comme la distribution de Φ et la répartition spatiale des sites des deux premières parties. Nous étudierons ensuite, les classes de sites de même codages qui, par construction, se retrouvent tous dans la même partie quelque soit la clé de tatouage utilisée. Nous constaterons que ces classes peuvent servir de point de départ à certaines attaques.

Afin de vérifier que les distributions observées suivent ou non certaines loi, nous aurons recours au test du χ^2. Nous commençons donc par rappeler l'utilisation de ce test statistique.

6.1 Validation des hypothèses de distribution

6.1.1 Test du χ^2

Dans cette section, nous utiliserons le test du χ^2 paramétré par un risque d'erreur α pour valider nos hypothèses sur certaines distributions. Ce test sera utilisé par exemple pour vérifier que la distribution de la proportion de sites qui vérifient Φ suit une loi normale.

Le test est basé sur le calcul de l'écart entre la distribution observée sur une expérience et celle attendue. Le test réussit si cet écart est assez petit (c'est-à-dire est inférieur à un seuil), sinon il échoue. La valeur du seuil est donnée par une table qui dépend du nombre de classes de la distribution (le degré de liberté) et du risque d'erreur α.

Lorsqu'on renouvelle l'expérience assez de fois et que la distribution observée suit bien la loi attendue, le test du χ^2 paramétré par un risque d'erreur $\alpha = 5\%$ échoue dans $\alpha = 5\%$ des cas en moyenne. Si ce n'est pas le cas, on peut conclure que la distribution observée ne suit pas la loi attendue. Notons que pour pouvoir appliquer le test, il faut que les cardinalités des classes de la distribution soient toutes supérieures à 5.

6.1.2 Distribution de la propriété Φ

Nous calculons la probabilité qu'un site satisfasse Φ, notée μ, pour un document non-tatoué. Afin d'avoir assez de documents, nous avons découpé les routes du Calvados selon une grille 20×20 et filtré les documents trop petits (moins de 100 sites). Pour chacun des 285 documents produits nous comptons le pourcentage de sites qui satisfont Φ. La figure 6.1 illustre le résultat de cette expérience. Sur cette figure, les pourcentages (en abscisse) sont échantillonnés par tranche de 1 point. L'axe des ordonnées représente le nombre de documents par pas d'échantillonnage. Pour rappel, nous avons tracé la loi normale $N(50, 2.55)$. La corrélation entre la distribution et cette loi normale, que l'on constate graphiquement, est validée par un test du χ^2 à 5%. Nous observons que la probabilité qu'un site extrait des tronçons de routes du Calvados satisfasse Φ suit une loi normale $N(50, 2.55)$.

La propriété Φ dépend de la perte de précision autorisée et du document testé. Nous avons renouvelé le test pour différentes valeurs de perte de précision et différents documents en retrouvant la même distribution avec toujours μ proche de $0,5$.

6.1.3 Répartition spatiale des sites

Pour savoir si le partitionnement est bien distribué dans le document, nous avons coloré les sommet centraux des sites du document qui sont impliqués dans le tatouage avec une clé fixée et une partition en 16 parties. Les figures 6.2(a) et 6.2(b) montrent les sites qui sont respectivement dans la première et la seconde partie de la partition. Nous avons coloré le sommet central de chaque site concerné. Nous voyons sur ces exemples que les sites sont bien répartis spatialement dans le document. Lorsqu'on renouvelle l'expérience avec d'autres clés, on constate visuellement le même phénomène.

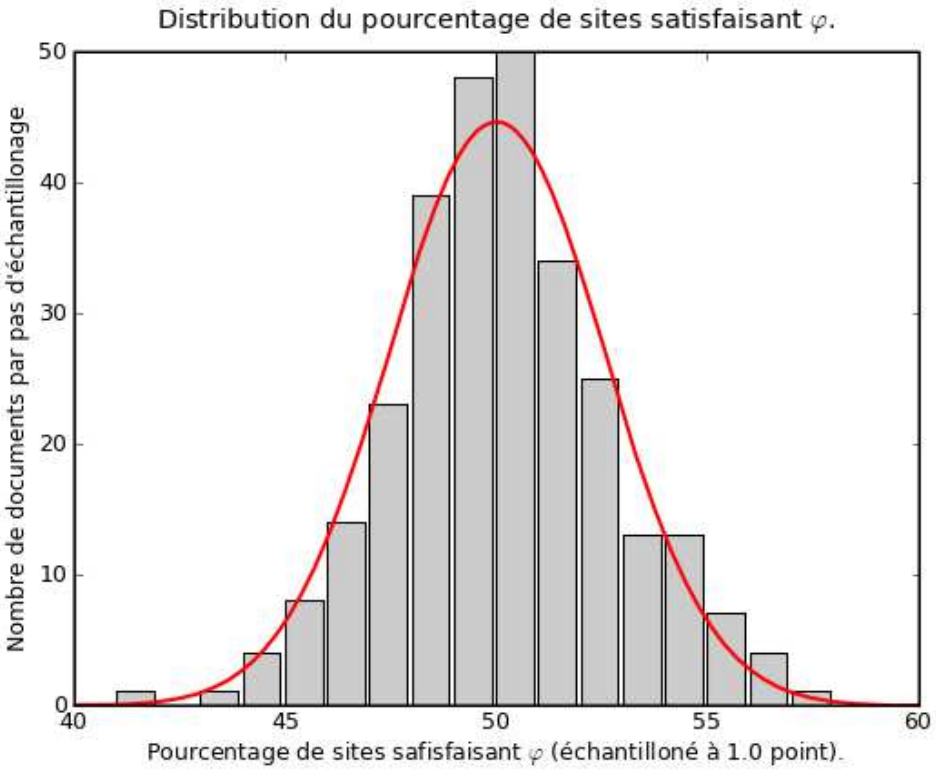


FIG. 6.1 – Distribution du pourcentage de sites satisfaisant Φ.

Les figures 6.3(a) et 6.3(b) sont des extraits respectifs des deux documents des figures précédentes. Remarquons que sur ces extraits, les sites sont toujours bien répartis spatialement.

C'est cette bonne répartition géographique des sites qui rend le schéma résistant au découpage. En effet, dans chaque extrait assez grand, on va retrouver suffisamment de sites dans chaque partie et reconstituer un sous-ensemble de la partition originale. Nous pourrons donc retrouver le biais introduit dans les deux premières parties.

Conclusion

Dans cette section, nous avons validé différents aspects qui contribuent au bon fonctionnement du schéma. Nous avons vu que la probabilité qu'un site choisi aléatoirement satisfasse Φ suit une loi normale dont nous avons donné les paramètres $N(50, 2.55)$. Par ailleurs, nous avons constaté que les sites sont bien répartis spatialement.

6.2 Problèmes de la corrélation des partitions pour deux clés différentes

Nous nous intéressons à la répartition des sites dans les différentes parties. Dans cette section, nous verrons que cette répartition n'est pas du tout uniforme. Selon la clé choisie, certaines parties peuvent contenir beaucoup plus de sites que d'autres. Une seconde expérience montrera que ce biais dans la répartition des sites entraîne une corrélation des partitions obtenues avec deux clés différentes.

6.2.1 Distribution des sites dans les parties

Pour évaluer la répartition des sites dans les parties, nous partitionnons les sites de chacun des 88 documents obtenus par une grille 10×10 en 12 parties. Pour chaque document, nous vérifions par un test du χ^2 à 5% si la distribution entre les parties est uniforme. Rappelons qu'un test du χ^2 à 5% doit réussir pour 95% des cas lorsque la distribution calculée suit bien la distribution attendue. On observe que le test du χ^2 réussit seulement pour 4 expériences sur 88. On peut donc conclure que la distribution des sites dans les parties n'est pas uniforme.

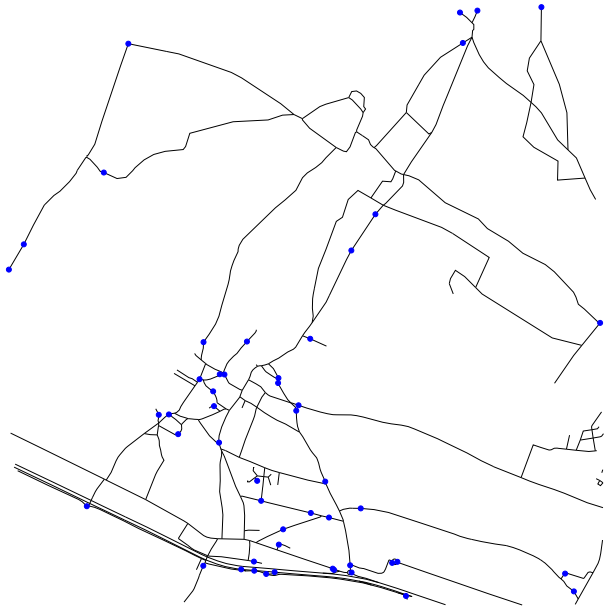
Cette conclusion s'explique par le fait que plusieurs sites peuvent avoir le même codage. Or, par construction de l'algorithme, tous les sites de même codage se retrouvent dans la même partie de la partition. Les expériences suivantes montreront que la cardinalité des classes de sites de même codage est variable. Ainsi, lorsque de grosses classes de sites de même codage se retrouvent dans la même partie, ils peuvent former une partie particulièrement grosse par rapport aux autres.

(a) Sommets centraux des sites appartenant à la première partie.

(b) Sommets centraux des sites de la deuxième partie.

FIG. 6.2 – Distribution géographique des sites de deux partitions pour les tronçons routiers du Calvados. (partition en 16 parties et clé fixée)

(a) Sommets centraux des sites appartenant à la première partie.



(b) Sommets centraux des sites de la deuxième partie.

FIG. 6.3 – Distribution géographique des sites de deux partitions sur un extrait des tronçons routiers du Calvados. (partition en 16 parties et clé fixée)

6.2.2 Corrélation des partitions pour deux clés différentes

Nous nous intéressons à la corrélation de la répartition des sites pour deux clés différentes. Notre protocole expérimental consiste à calculer le partitionnement obtenu avec la seconde clé pour les sites qui se trouvent dans la première partie lors du partitionnement obtenu avec la première clé. Nous vérifions si les sites sont bien répartis uniformément dans les parties à l'aide d'un test du χ^2 à 5%.

Sur notre corpus de documents issus du découpage selon la grille 10×10, en prenant un partitionnement en 8 parties et en fixant deux clés, le test χ^2 trouve une répartition uniforme pour moins de 20% des expériences. Pour cette expérience, nous devons donc rejeter l'hypothèse de distribution uniforme des sites dans les parties. En renouvelant l'expérience avec d'autres paramètres, nous aboutissons toujours à la même conclusion : les partitionnements de sites pour deux clés différentes sont corrélés.

Conclusion

Nous savons que tous les sites de même codage sont traités de la même façon par le schéma. Les expériences précédentes ont montré qu'à cause de cela, les partitions obtenus par deux clés différentes sont corrélées. Un utilisateur malveillant pourrait donc regarder les grosses classes de sites de même codage pour estimer celles qui portent la marque. En partant de cette information, l'utilisateur peut vérifier si un document est tatoué, il pourrait même laver la marque.

Dans la section suivante, nous allons étudier les classes de sites de même codage. Nous verrons, entre autres, que les répartitions des classes de codage dans les parties ne sont pas corrélées pour deux clés différentes. Cela nous permettra de construire une variante du schéma pour laquelle ces attaques ne sont pas applicables. Cette variante consiste à ne traiter qu'un seul site par classe de codage.

6.3 Étude des classes de codage

Nous avons vu dans la section précédente que le schéma peut poser des problèmes de sécurité. Dans le chapitre suivant, nous proposerons une variante pour y remédier. Cette section a pour objectif de mieux comprendre les problèmes liés aux grosses classes de codages. Nous définissons une *classe de codage* comme l'ensemble des sites qui ont le même codage.

Nous commencerons par montrer que les classes de codage sont distribuées uniformément dans les parties. Nous verrons ensuite que l'on dispose de beaucoup de codages différents par rapport au nombre de sites. Ces deux résultats permettront de montrer que l'on a assez de sites dans chaque partie pour pouvoir effectuer le tatouage. Une troisième expérience montrera que beaucoup de classes de codages sont des singletons ou de petites tailles. Enfin, nous constaterons que le partitionnement des classes de codage est uniforme et que les partitions obtenus par deux clés différentes sont indépendantes.

6.3.1 Distribution des classes de codages dans les parties

Dans cette expérience nous allons vérifier que la distribution des classes de codages est uniforme dans les parties. Pour cette expérience, nous partitionnons tous les sites d'un document en ne gardant qu'un seul représentant par classe de codage.

Le protocole expérimental consiste à partitionner le document en 12 parties, les sites de chacun des 88 documents non-vides des tronçons de routes du Calvados découpés selon une grille de 10×10. Comme pour l'expérience présentée dans la section 6.2.1, pour chaque document nous vérifions si la distribution entre les parties est uniforme par un test du χ^2 à 5%. Le test réussi pour 87 expériences sur 88. On en conclut que les groupes de sites de même codage sont distribués uniformément entre les parties.

Ce résultat n'est pas étonnant car la répartition des classes de codage dans les parties, pour une clé fixée, dépend uniquement du codage des sites. Or, pour cette expérience, nous avons partitionné des codages qui sont tous différents. Et, par définition de la fonction de hachage, deux entrées différentes vont donner deux sorties indépendantes. Les parties assignées à deux codages différents sont donc indépendantes. Pour toute clé, un codage choisi aléatoirement parmi ceux des sites d'un document a donc autant de chances de se retrouver dans n'importe quelle partie de la partition. La distribution des codages dans les parties est donc uniforme.

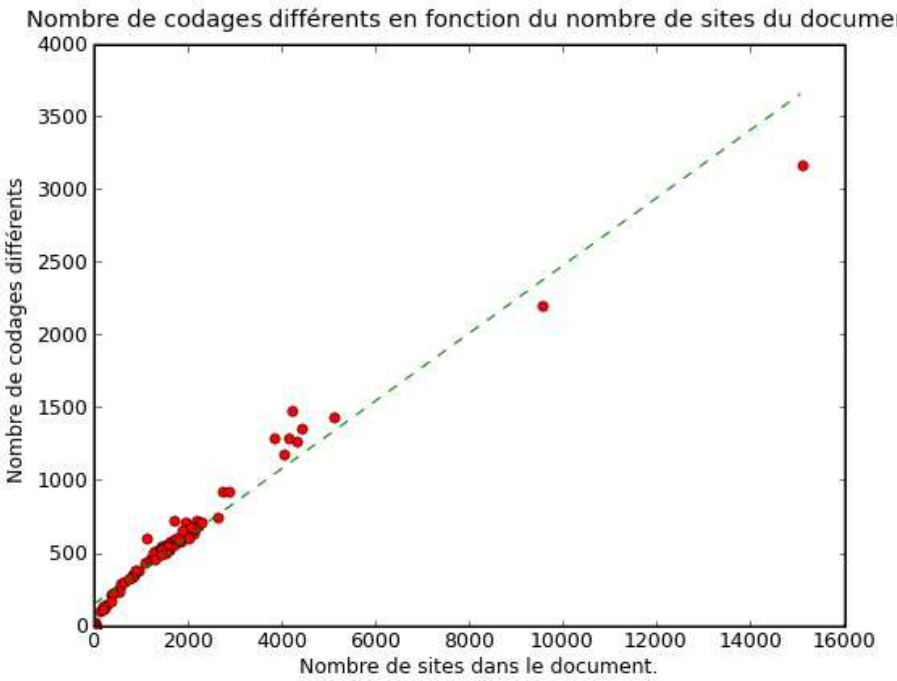
6.3.2 Relation entre nombre de sites et nombre de classes de codage

Les expériences précédentes ont montré que les classes de codages sont partitionnées uniformément. En partant de ce résultat, nous allons montrer que, même si les sites ne sont pas distribués uniformément dans les parties, on dispose tout de même d'un minimum de sites dans chaque partie.

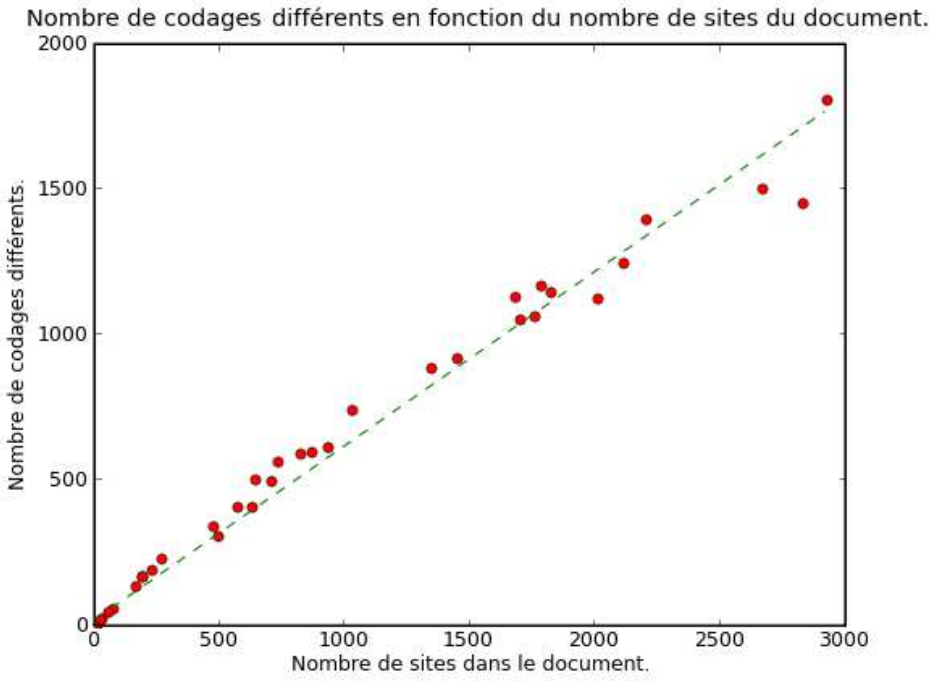
L'expérience présentée dans cette section montre l'évolution du nombre de codages différents en fonction de la taille du document. Cette expérience consiste à compter le nombre de codages différents pour des documents de différentes tailles. La figure 6.4(a) montre le résultat de cette expérience pour les documents issus du découpage des tronçons de routes du Calvados selon une grille 10×10, 6×6 et 3×3. Chaque document du corpus est représenté par un point dont l'abscisse et l'ordonnée correspondent respectivement au nombre de sites et codages différents dans le document. Sur cette figure, on voit que le nombre de codages différents croît avec le nombre de sites du document. On remarque aussi que cette augmentation a tendance à devenir de plus en plus faible à mesure que le nombre de sites augmente.

En s'intéressant plus particulièrement aux documents de moins de 3000 sites (figure 6.4(b)), nous remarquons que le nombre de codages différents est quasiment linéaire en le nombre de sites pour les petits documents. Sur cet exemple, le coefficient de corrélation linéaire est de 99% avec la droite $y = 0,6x$. En d'autres termes, pour les documents de moins de 3000 sites, on a de l'ordre de 60% de codages différents par rapport au nombre de sites.

On constate donc qu'un document contient beaucoup de classes de codages. Or, nous avons vu que les classes de codages sont partitionnées uniformément. On en conclut que nous aurons



(a) Pour l'ensemble des documents de moins de 16000 sommets



(b) Pour les documents de moins de 3000 sommets

FIG. 6.4 – Nombre de codages différents en fonction du nombre de sites dans le document.

suffisamment de sites dans chaque partie.

6.3.3 Distribution des cardinalités des classes de codages

Nous souhaitons maintenant connaître la distribution des cardinalités des classes de codages. Nous voulons connaître la proportion de singletons et de classes comprenant beaucoup de sites. Pour cette expérience, nous avons calculé la proportion des classes de codages en fonction de leur cardinalité. Pour les documents du corpus des tronçons de routes découpé selon une grille 10×10, nous avons représenté la proportion cumulés de classes de codage différentes en fonction de leur cardinalité. L'histogramme de la figure 6.5(a) présente le résultat de cette expérience. Il donne le pourcentage de classes de codage ayant au plus un certain nombre de représentants. La position en abscisse de chaque barre donne la cardinalité x maximum de la classe de codages. La hauteur de la barre représente la proportion moyenne sur toutes les expériences (ainsi que l'écart type) des classes de codages ayant au plus x représentants.

Sur ce graphique, le pourcentage moyen de classes de codages singletons (qui ne comportent qu'un site) est de 70% pour un écart type de 6 points. Ces classes sont donc largement majoritaires dans le corpus issu d'un découpage des tronçons de routes selon une grille de 10×10. Par ailleurs, nous constatons que, pour ce corpus, 90% des classes de codages ont moins de 5 sites.

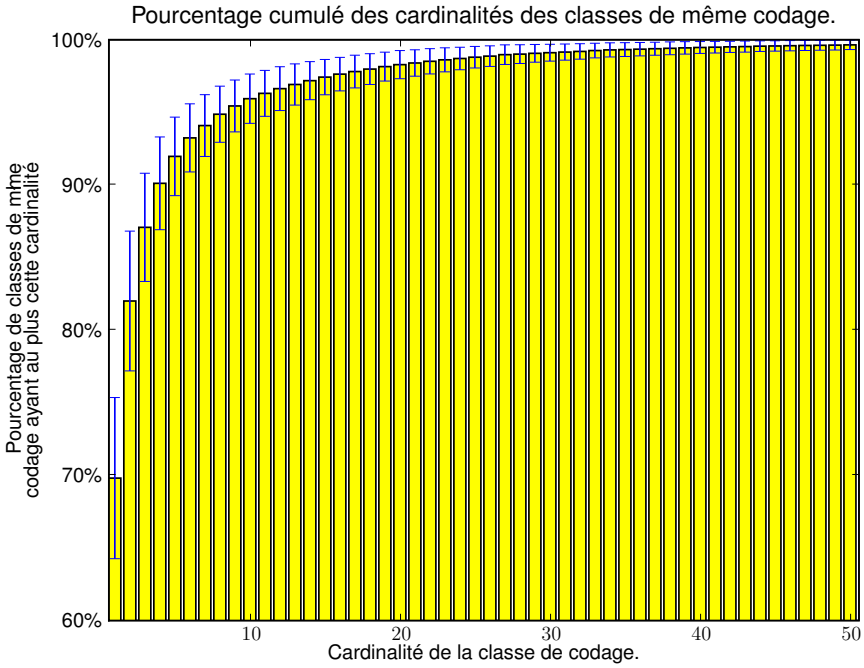
La figure 6.5(b) montre le résultat de la même expérience, cette fois ci sur le corpus issu des tronçons de routes du Calvados découpés selon une grille de 3×3. Pour cette expérience, nous avons moins de documents et chaque document contient beaucoup plus de sommets. Sur cette figure, le pourcentage de classes singletons diminue légèrement (65%). Nous constatons aussi que l'écart type diminue, les valeurs calculées sont donc beaucoup plus homogènes. Enfin, nous voyons qu'environ 5% des classes de sites de même codage contiennent plus de 50 sites.

En comparant les deux figures, on remarque que la proportion de petites classes de codages diminue et que de grosses classes apparaissent quand le nombre de sites du document augmente. La présence de ces grosses classes explique que le partitionnement des sites ne soit pas uniforme. En effet, par construction du schéma, tous les sites d'une même classe de codage se retrouvent dans la même partie. Si plusieurs grosses classes de codage se retrouvent dans la même partie, la cardinalité cette partie va être beaucoup plus importante que celle des autres.

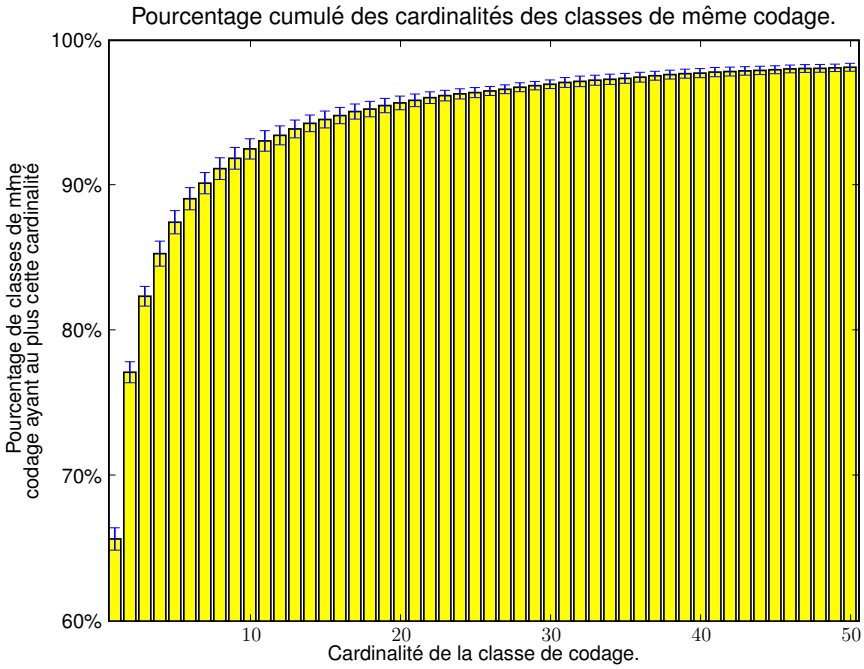
On voit cependant que la plupart des classes contiennent peu de sites. Dans la seconde expérience par exemple : 65% des classes de codages sont des singletons.

6.3.4 Corrélation des partitions des classes de codages pour deux clés différentes

Cette expérience vise à vérifier que les distributions des classes de codage dans les parties avec deux clés différentes sont bien indépendantes. Pour ce test, nous reprenons le protocole expérimental de l'expérience 6.2.2 en ne conservant qu'un seul représentant (choisi aléatoirement) par classe de codage. Pour rappel, cette expérience consiste à faire une première partition avec une clé, puis à vérifier que les sites de la première partie sont bien distribués uniformément dans les parties quand on tatoue avec une seconde clé. Nous vérifions l'uniformité de la distribution



(a) Pour le corpus des tronçons de routes découpés selon une grille 10×10.



(b) Pour le corpus des tronçons de routes découpés selon une grille 3×3.

FIG. 6.5 – Pourcentage cumulé des cardinalités des classes de sites de même codages.

par un test du χ^2.

L'expérience est réalisé sur un corpus obtenu par découpage du document original (routes du Calvados) selon une grille 6×6. Nous ne retenons que 30 des 36 documents obtenus (nous avons écarté 6 documents trop petits pour pouvoir exécuter le test du χ^2). On fixe le nombre de parties à 8.

Cette fois, le test du χ^2 réussit pour 29 des 30 documents. Nous concluons donc que les distributions des codages pour les deux clés différentes sont indépendantes, pour cette expérience. En renouvelant l'expérience sur différents corpus et en faisant varier les paramètres, on arrive à la même conclusion : les répartitions des codages dans les parties pour deux clés différentes sont indépendante.

Ce résultat s'explique par le fait que la distribution des codages dans les parties est donnée par le hachage du codage et de la clé. Étant donné qu'on ne traite jamais deux fois le même codage, à clé fixée, on ne hache jamais deux fois la même valeur, et, *a fortiori*, pour deux clés différentes, on ne hache que des valeurs différentes. Or, pour une fonction de hachage bien choisie, deux entrées différentes donnent deux hachés décorrélés. Par conséquent chaque haché est distribué dans les parties indépendamment des autres. Il est donc normal que les partitionnements pour deux clés différentes soit indépendants.

Conclusion

Cette section a montré que les classes de codages sont distribuées uniformément dans les parties. Par ailleurs, nous avons vu que le nombre de classes de codages est relativement important par rapport au nombre de sites du documents. Ces deux observations permettent de conclure que nous aurons assez de sites dans chaque partie.

Nous avons montré que la plupart des classes de codages sont des singletons et qu'il peut exister de grosses classes de codages. Ces grosses classes sont à l'origine des attaques potentielles qui pourraient permettre de détecter la marque, laver le document ou retrouver la clé qui a permis de tatouer le document. En effet, si une grosse classe est impliquée dans le tatouage elle va porter un biais statistique. Dans ce cas, il est possible de retrouver celui-ci et de voir que le document est bien tatoué sans avoir besoin de clés. Si plusieurs grosses classes sont impliquées dans le marquage, il est d'autant plus facile de décider si un document est taoué ou non. Le schéma que nous avons proposé n'est donc pas sûr.

Conclusion de l'étude statistique

Les études présentées dans ce chapitre nous ont donné une certaine expertise du schéma. Nous avons conçu des expériences afin de vérifier que la propriété Φ suit bien une loi normale, et que les sites de chaque partie de la partition sont bien répartis spatialement dans le document. Nous avons aussi obtenu beaucoup d'information sur la distribution des sites et des classes de codages dans les parties. Tous ces protocoles expérimentaux pour évaluer le schéma pourront

être repris dans le cadre du schéma générique présenté dans la partie III.

Dans ce chapitre, nous avons vu que les grosses classes de codages peuvent être à l'origine d'attaques pour laver le document par exemple. Cependant, cette étude des classes de codages va aussi nous permettre de construire une variante du schéma qui ne soit pas sensible aux attaques basées sur les grosses classes de codage. Le chapitre suivant présentera cette variante et en fera l'étude.

Chapitre 7

Variantes du schéma

Sommaire

Dans ce chapitre, nous présentons deux variantes de l'algorithme. La première vise à améliorer la robustesse du schéma face au découpage. Elle consiste à filtrer les sites qui se trouvent sur l'enveloppe convexe, qui sont le plus susceptible de changer de codage lors d'un découpage.

La seconde variante permet de résoudre le problème des grosses classes de codages, point de départ d'éventuelles attaques, en ne tenant compte que d'un seul site par classe de codage. Nous verrons que cette variante demande des documents plus gros pour fonctionner correctement.

Ces deux variantes peuvent être combinées afin d'améliorer la robustesse au découpage et de se prémunir contre le problème des grosses classes de codages.

7.1 Filtrage des sites de l'enveloppe convexe

L'opération de découpage modifie le voisinage des sites de l'enveloppe convexe. Sur les figures 7.1 et 7.2, les sites concernés sont encerclés en rouge. D'après la définition du schéma, le codage de ces sites et le fait qu'ils respectent ou non Φ ont donc toutes les chances d'être altérés. Ces sites deviennent alors un bruit pour le schéma. En les filtrant (c'est-à-dire en les ignorant), nous traitons moins de sites. Cependant, quand le document a subi un découpage, le biais de l'ensemble des sites que nous traitons sera plus important. Nous pourrons donc retrouver la marque dans des échantillons de documents plus petits.

FIG. 7.1 – Enveloppe convexe de routes du Calvados. Les sommets entourés en rouge font partie de l'enveloppe convexe.

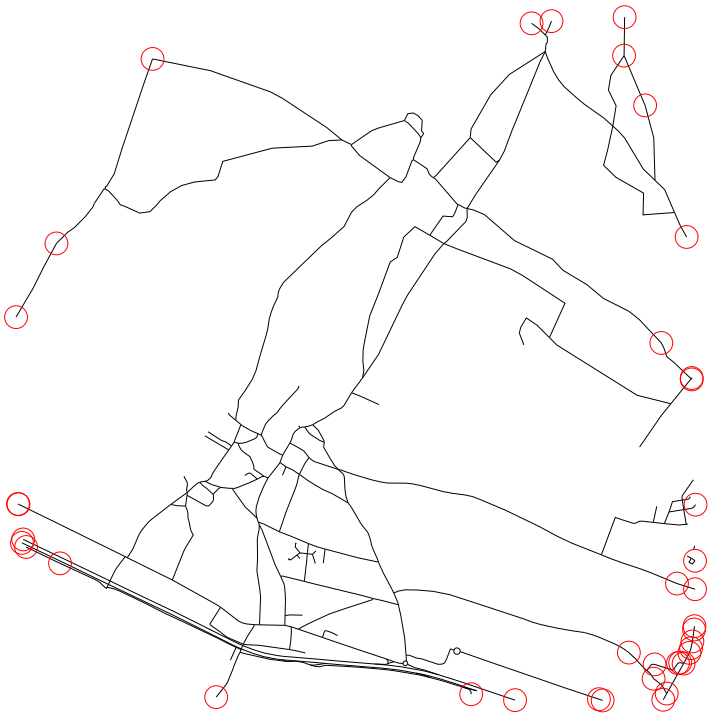


FIG. 7.2 – Enveloppe convexe d'un échantillon des routes du Calvados. Les sommets entourés en rouge font partie de l'enveloppe convexe.

Dans cette section, nous montrons l'impact du filtrage des sites de l'enveloppe convexe sur la détection de la marque quand le document n'a pas subi de modification, quand il a été découpé et, enfin, quand il a été retatoué.

7.1.1 Détection de la marque

Nous commençons par vérifier l'impact du filtrage des sites de l'enveloppe convexe sur la détection de la marque.

La figure 7.3 montre le résultat de l'expérience avec un partitionnement en 4 parties et une perte de précision autorisée de 1 mètre sur le corpus des routes du Calvados découpé selon une grille de 10×10. Le tableau 7.1 donne une synthèse du tableau. On constate que la variante influe peu sur la détection. Les résultats sont comparables à ceux présentés dans la section 5.3.

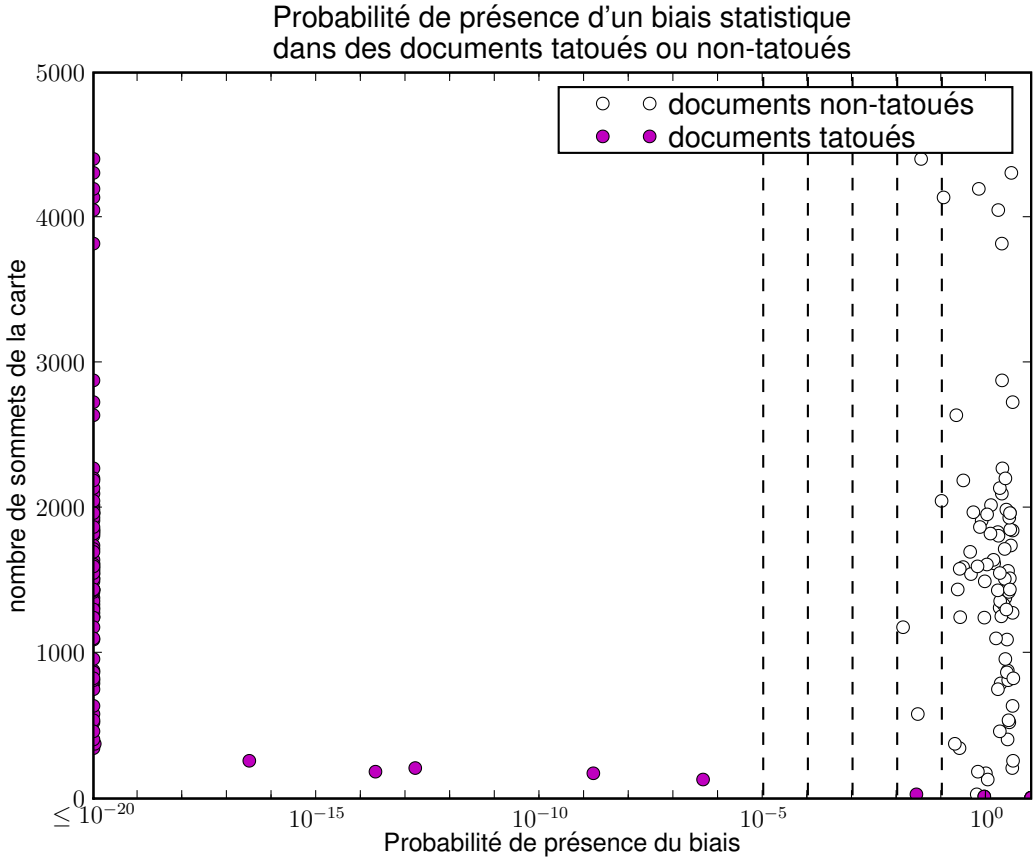


FIG. 7.3 – Résultat de la détection sur le corpus en filtrant les sites de l'enveloppe convexe.

Lorsque nous renouvelons l'expérience sur plusieurs corpus en faisant varier la perte de précision autorisée et le nombre de parties de la partition, on obtient le tableau 7.4. En comparant ce tableau à celui obtenu sans filtrer les sites de l'enveloppe convexe (figure 5.6), on

Seuil	Documents de plus de :					
	0 sommet		100 sommets		200 sommets	
	88 documents		84 documents		71 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	4	3	4	0	4	0
10^{-2}	0	4	0	0	0	0
10^{-3}	0	4	0	0	0	0
10^{-4}	0	4	0	0	0	0
10^{-5}	0	4	0	0	0	0

TAB. 7.1 – Résultat de la détection sur le corpus de test en filtrant les sites de l'enveloppe convexe.

peut confirmer l'observation précédente : le filtrage des sites de l'enveloppe convexe n'a que peu d'influence sur la détection de la marque.

7.1.2 Robustesse au découpage

L'expérience précédente a montré que le tatouage fonctionne toujours, même en filtrant les sites de l'enveloppe convexe. Nous vérifions maintenant si cette variante permet de mieux retrouver la marque dans des documents plus petits obtenus par découpage d'un document tatoué.

En fixant la perte de précision autorisée à 1 mètre, le nombre de parties à 4 et une clé, nous obtenons le résultat illustré par la figure 7.4 et la table 7.2. Ce tableau montre que les résultats sont légèrement meilleurs pour les très petits documents (moins de 200 sommets) que le schéma initial (pour les faux négatifs).

Seuil	Documents de plus de :					
	0 sommets		200 sommets		400 sommets	
	289 documents		209 documents		164 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	3	38	2	0	1	0
10^{-2}	0	52	0	1	0	0
10^{-3}	0	61	0	2	0	0
10^{-4}	0	72	0	4	0	0
10^{-5}	0	81	0	10	0	1

TAB. 7.2 – Résultat de la détection après découpage en filtrant les sites de l'enveloppe convexe.

Ce résultat est confirmé par le tableau 7.5 (page 115) obtenu en faisant varier la perte de précision autorisée et le nombre de parties. Les cases sont colorés en rouge (resp. en vert) si leur

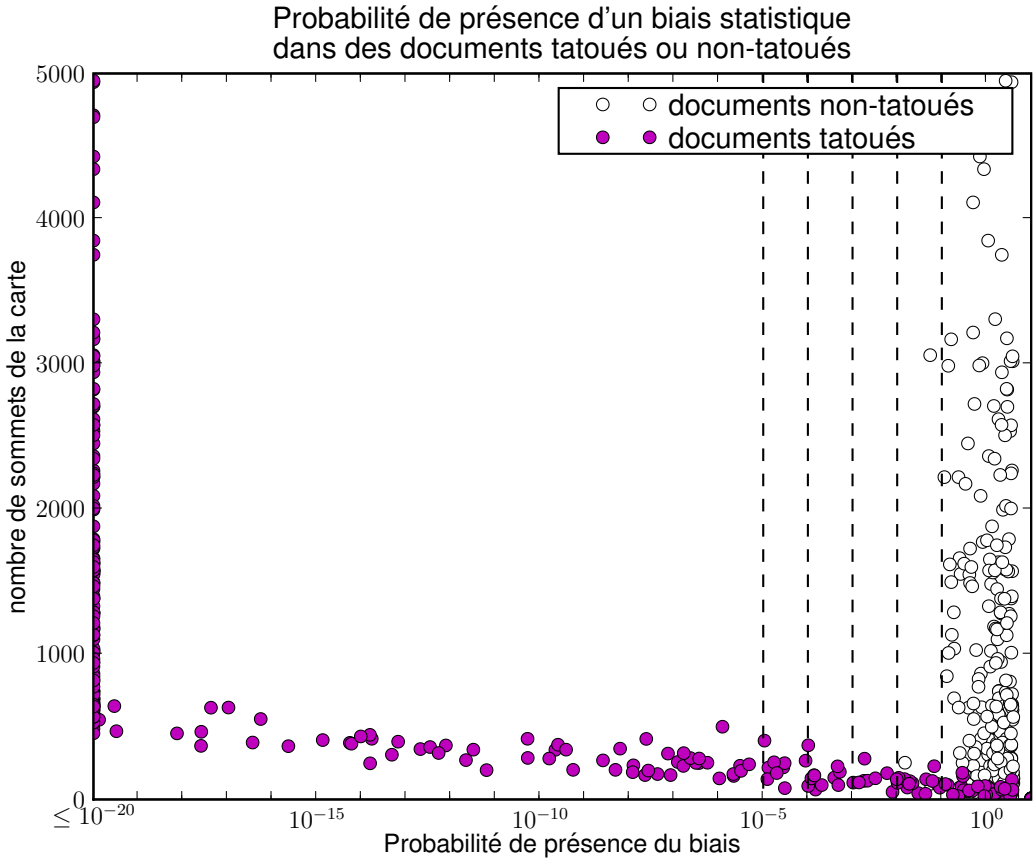


FIG. 7.4 – Résultat de la détection après découpage en filtrant les sites de l'enveloppe convexe.

valeur est supérieure (resp. inférieure) de plus de 15% au tableau de référence 5.8 (page 87). La comparaison avec le tableau de référence montre clairement l'amélioration de la détection pour les petits documents lorsque la perte de précision autorisée est faible (1 mètre). En considérant uniquement une perte de précision autorisée de 1 mètre, nous comptons 9 cas pour lesquels la détection est meilleure contre 1 cas pour lequel la détection est moins bonne.

7.1.3 Robustesse au retatouage

Pour valider la robustesse du schéma contre le retatouage, nous réitérons l'expérience de la section 5.4.3 en filtrant les sites de l'enveloppe convexe. Pour les mêmes paramètres que l'expérience 5.4, c'est-à-dire une partition en 4 parties, une perte de précision autorisée de 1 mètre et en fixant deux clés, nous obtenons les résultats illustrés par la figure 7.5 et le tableau 7.3. Quand on les compare au tableau 5.4, on voit que quelques faux-négatifs apparaissent pour un seuil de 10^{-3} pour des documents ayant entre 200 et 400 sommets. Nous en concluons que, pour l'expérience considérée, le filtrage des sites de l'enveloppe convexe nuit peu à la détection de la première marque après retatouage.

Le tableau 7.3 synthétise les résultats lorsqu'on fait varier la perte de précision autorisée et le nombre de parties. Globalement, ces tableaux font apparaître 20 cas pour lesquels on voit des améliorations et 22 cas pour lesquels la détection est moins bonne. Par conséquent, sur l'ensemble des expériences, le filtrage des sites de l'enveloppe convexe n'a que peu d'influence sur la robustesse du schéma face au retatouage.

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	88 documents		81 documents		77 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	3	5	3	0	3	0
10^{-2}	2	6	2	1	2	0
10^{-3}	1	7	1	2	1	0
10^{-4}	1	10	1	4	1	1
10^{-5}	0	12	0	5	0	2

TAB. 7.3 – Résultat de la détection après découpage en filtrant les sites de l'enveloppe convexe.

Conclusion du filtrage des sites de l'enveloppe convexe

Dans cette section, nous avons évalué une variante du schéma qui consiste à filtrer les sites de l'enveloppe convexe. Nous savons que ces sites sont les plus susceptibles d'être altérés par le découpage de la carte. En ne tenant pas compte de ces sites, nous espérons pouvoir retrouver la marque dans des extraits de documents plus petits.

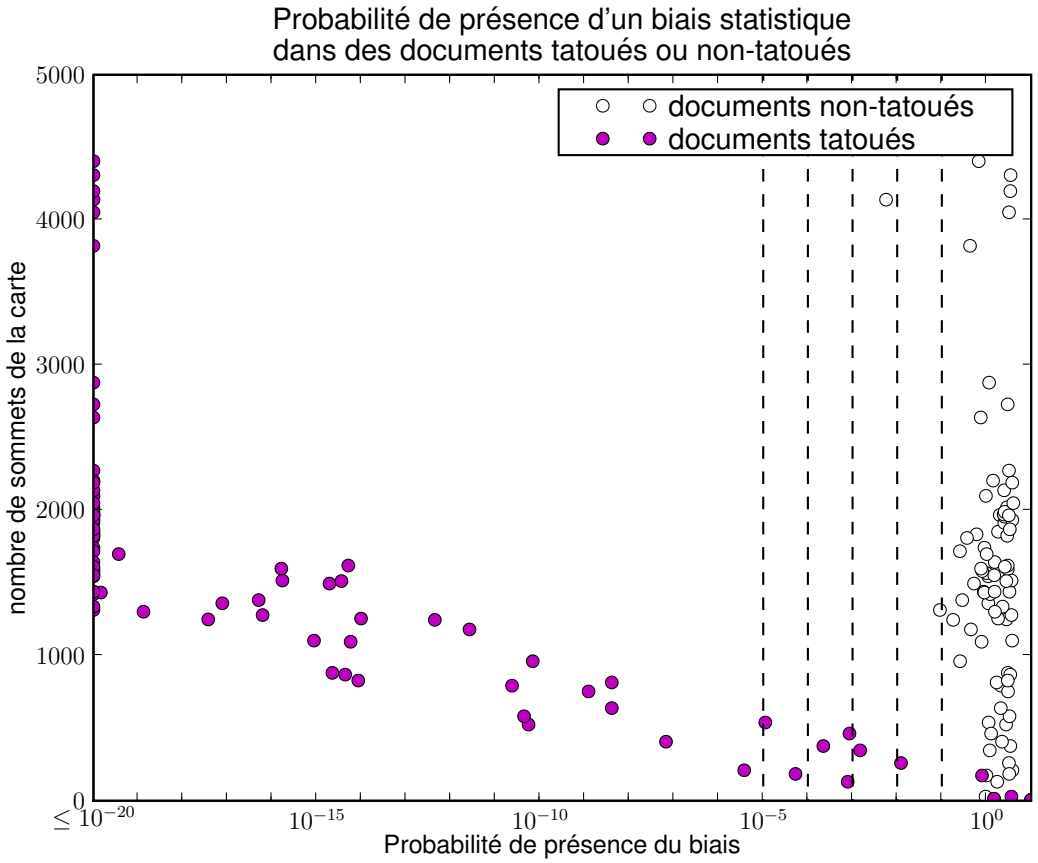


FIG. 7.5 – Résultat de la détection après retatouage sur le corpus en filtrant les sites de l'enveloppe convexe.

Nous avons vu que cette variante nuit peu à la détection de la marque et sur le retatouage. En revanche, les résultats sont meilleurs quand le document a été découpé. Pour une perte de précision autorisée de 1 mètre, nous avons constaté une très nette amélioration des résultats.

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	14	29	80	80	80	80	29	29	80	80	185	155	29	80	131	185	210	311
10^{-2}	29	29	80	155	210	155	29	29	155	174	347	377	29	80	377	377	525	377
10^{-3}	29	29	80	185	210	174	29	80	174	185	347	377	80	185	377	377	881	539
10^{-4}	29	80	174	185	210	210	80	155	185	377	347	377	80	185	377	377	881	793
10^{-5}	29	80	185	281	377	311	80	174	311	377	881	377	174	185	377	881	1255	1337

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	34	49	103	99	174	103	49	103	174	165	236	236	103	103	236	228	228	477
10^{-2}	49	49	103	103	199	236	103	103	236	174	278	477	103	174	477	477	906	800
10^{-3}	49	103	103	198	202	236	103	174	278	566	278	477	103	236	800	558	998	800
10^{-4}	49	103	202	198	278	278	103	174	278	566	564	477	174	236	800	800	1027	948
10^{-5}	103	103	236	236	353	278	174	236	477	566	1034	694	236	236	800	800	1232	1034

TAB. 7.4 – Taille du plus grand document pour lequel la détection de la marque a échoué en filtrant les sites de l'enveloppe convexe (Les corpus issus du même document original sont fusionnés, chaque expérience est renouvellée avec 3 clés différentes).

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	161	184	373	267	501	444	229	281	418	501	501	554	280	372	632	444	665	632
10^{-2}	281	281	373	372	501	501	280	502	502	501	1177	820	502	502	632	554	1011	1011
10^{-3}	281	316	501	501	571	554	343	502	502	646	1177	1011	502	502	647	867	1177	1011
10^{-4}	281	373	554	632	661	554	343	502	554	647	1177	1131	502	632	1011	967	1737	1578
10^{-5}	281	418	554	632	849	571	502	502	1011	666	1262	1131	502	632	1011	967	4428	2220

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	162	215	263	283	403	386	211	263	423	423	772	452	373	373	969	429	767	969
10^{-2}	162	294	294	403	423	398	263	372	647	486	772	647	373	423	969	969	812	969
10^{-3}	211	372	403	423	631	452	373	452	647	767	814	807	373	647	969	969	969	1139
10^{-4}	263	403	423	441	631	582	373	452	767	969	814	969	374	647	969	969	1219	1350
10^{-5}	263	403	423	490	772	767	402	452	767	969	969	969	423	768	969	1012	1359	1350

TAB. 7.5 – Taille du plus grand document pour lequel la détection de la marque a échoué après découpage en filtrant les sites de l'enveloppe convexe
(Les corpus issus du même document original sont fusionnés, chaque expérience est renouvellée avec 3 clés différentes).

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	9017	260	174	185	260	155	4232	377	260	210	347	260	2205	377	377	377	881	407
10^{-2}	15399	260	260	260	347	347	4763	377	463	525	881	463	2872	377	377	377	881	793
10^{-3}	15399	347	347	311	377	407	5245	463	539	525	881	463	4699	525	793	881	1719	793
10^{-4}	18554	463	463	407	539	463	9017	525	582	539	881	869	4699	869	1311	881	1846	881
10^{-5}	18554	539	463	407	1255	463	9017	815	881	753	1440	881	4699	881	1311	1594	2640	1311

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	5620	199	198	353	353	278	2443	797	477	800	353	477	1311	766	800	903	1027	800
10^{-2}	6134	400	278	859	353	477	2966	797	786	981	694	948	2354	1311	1017	1074	1142	1034
10^{-3}	7814	676	477	859	694	477	2967	906	948	1238	1034	948	2667	1311	1142	1299	1232	2442
10^{-4}	8122	928	657	1034	723	477	3436	981	948	1311	1269	948	2733	1311	1299	1955	2083	2442
10^{-5}	8122	928	694	1235	849	684	6829	981	948	1311	1269	1074	3436	1311	2442	2745	2083	2442

TAB. 7.6 – Taille du plus grand document pour lequel la détection de la marque a échoué après retatouage en filtrant les sites de l'enveloppe convexe

7.2 Modification d'un seul site par classe de codages

Dans cette section, nous reprenons les résultats précédents et construisons une variante de l'algorithme insensible aux attaques basées sur l'analyse des grosses classes de codages. Pour cette variante, au lieu de traiter tous les sites d'une même classe de codage, nous ne traiterons qu'un seul site par classe. Cela réduit bien sûr énormément la quantité de sites traités, la marque sera donc moins présente dans le document. Cependant, avec cette variante, il n'existe plus de grosses classes de codage qui sont le point de départ d'éventuelles attaques par étude des grosses classes de codage.

En pratique, nous itérons aléatoirement sur l'ensemble des sommets du document. Avant de traiter un site avec la chaîne de traitement précédente, nous vérifions si un site de même codage a déjà été traité. Si c'est le cas, nous abandonnons le traitement du site et passons au suivant.

L'expérience 6.3.2 a montré la relation presque linéaire qui existe entre le nombre de sites d'un document et le nombre de classes de codages. D'après cette expérience, on dispose déjà de beaucoup de classes de codages dans les petits documents et cette quantité croît avec le nombre de sites du document. Nous disposerons donc d'assez de sites pour marquer le document.

Avec notre variante, pour chaque classe de codages de taille n, l'algorithme de détection a donc 1 chance sur n de choisir le site qui a été marqué lors du tatouage. Dans le cas particulier des singletons, on est certain de retrouver le bon site. Par conséquent, la prépondérance des singletons, montrée par l'expérience 6.3.3, nous garantit de retrouver une grande partie des sites marqués lors de la phase de tatouage.

Dans cette section, nous vérifierons que la détection de la marque est toujours possible avant d'étudier la robustesse de ce nouveau schéma contre le découpage et le retatouage.

7.2.1 Détection de la marque

Nous souhaitons vérifier que la détection fonctionne toujours pour cette variante. Nous reprenons le protocole expérimental de l'expérience 5.3. Pour rappel, cette expérience consiste à appliquer l'algorithme de détection sur chaque document du corpus tatoué et non-tatoué.

La figure 7.6 et le tableau 7.7 donnent le résultat de l'expérience pour le corpus des tronçons de routes du Calvados découpé selon une grille 10×10, avec une perte de précision autorisée de 1 mètre et une partition en 4 parties. Ces figures montrent que la marque est moins présente dans le document qu'avec la version originale du schéma. Cependant, on voit clairement que nous distinguons les documents tatoués de ceux qui ne le sont pas.

Le tableau 7.10 (page 123) donnent le plus grand document pour lequel l'expérience a échoué pour différents nombres de parties et différentes valeurs de perte de précision autorisée. Sur ce tableau, nous pouvons constater que la détection de la marque nécessite des documents de plus en plus gros (en nombre de sites) à mesure que le nombre de parties augmente. Ce qui est normal car seuls les sites des deux premières parties de la partition portent la marque, plus le nombre de parties augmente, plus la proportion de sites concernés est faible. Pour fonctionner correctement, l'algorithme de détection nécessite d'avoir un minimum de sites concernés par le tatouage. Pour

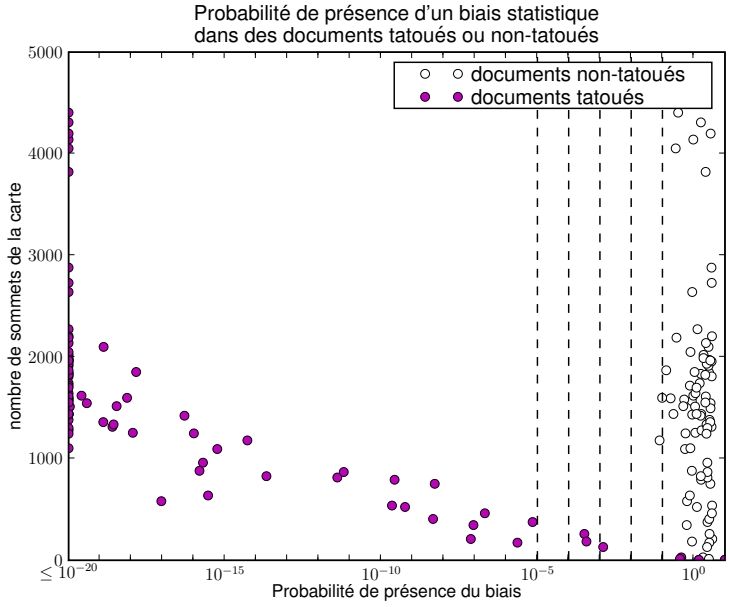


FIG. 7.6 – Résultat de la détection sur le corpus en ne prenant qu'un seul site par classe de codage.

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	88 documents		81 documents		77 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	2	4	2	0	2	0
10^{-2}	0	4	0	0	0	0
10^{-3}	0	5	0	0	0	0
10^{-4}	0	7	0	1	0	0
10^{-5}	0	7	0	1	0	0

TAB. 7.7 – Résultat de la détection sur le corpus de test en ne prenant qu'un site par classe de codage.

atteindre ce minimum, le document doit contenir assez de sommets. Il est donc normal que la détection de la marque demande des documents plus gros quand augmente le nombre de parties de la partition. On remarque aussi que plus la perte de précision autorisée augmente, plus l'algorithme de détection nécessite de gros documents. Lorsque la perte de précision autorisée augmente, le nombre de modifications de sites annulées lors du tatouage augmente aussi. En effet, plus on bouge le sommet central du site, plus on risque de le faire sortir des cercles circonscrits définis dans la section 4.3.2, et donc de devoir annuler le déplacement.

7.2.2 Robustesse au découpage

Pour vérifier que l'algorithme est toujours robuste au découpage, nous reprenons le protocole expérimental de la section 5.4.2. Pour mémoire, ce protocole expérimental consiste à découper le document original selon une grille 4×4. Les documents obtenus sont tatoués et nous exécutons l'algorithme de détection sur 20 rectangles découpés aléatoirement dans chacun d'eux.

Les résultats de l'expérience, pour une partition en 4 parties et une perte de précision autorisée de 1 mètre, sont donnés par la figure 7.7. Le tableau 7.8 en donne une synthèse. Nous constatons que le schéma requiert des extraits plus grands que précédemment pour bien fonctionner. Sur la figure, on remarque que pour être sur de distinguer les documents tatoués ou non il faut que les extraits découpés contiennent au moins 2000 sommets.

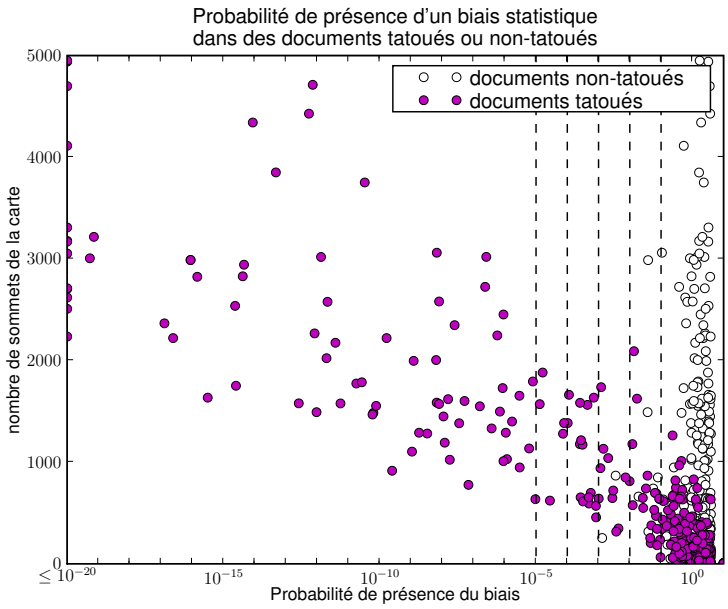


FIG. 7.7 – Résultat de la détection après découpage du document sur le corpus en ne prenant qu'un seul site par classe de codage.

Ce résultat est confirmé par le tableau 7.11 (page 124). Lorsqu'on fait varier le nombre de parties et la perte de précision autorisée, on constate que les résultats sont beaucoup moins bons

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	289 documents		209 documents		164 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	9	148	9	70	7	32
10^{-2}	2	168	2	88	1	45
10^{-3}	0	179	0	99	0	54
10^{-4}	0	194	0	114	0	69
10^{-5}	0	199	0	119	0	74

TAB. 7.8 – Résultat de la détection sur le corpus de test en ne prenant qu'un site par classe de codage.

que ceux présentés dans le tableau 5.8 (page 87).

Ce résultat s'explique par le choix du site qui est fait aléatoirement parmi ceux de sa classe de codage. Certains sites choisis lors du tatouage ne font pas partie de l'extrait découpé. Dans ce cas, nous n'avons aucune chance de les retrouver lors de la détection. Lorsqu'on s'intéresse au cas particulier de singletons, qui, comme nous l'avons vu précédemment forment la plupart des classes de codages, deux cas de figure apparaissent. Les singletons présents dans l'extrait qui n'en étaient pas dans le document original risquent de nuire à la détection. Par contre, ceux qui étaient présent dans l'extrait qui étaient déjà des singletons dans le document original vont aider à la détection de la marque.

En fait, les seuls sites marqués que nous sommes certains de retrouver lors de la détection sont ceux qui formaient des classes de codages singletons dans le document original. Ceux-ci donnent un biais statistique plus faible que dans la version originale du schéma. Nous pouvons retrouver ce biais, même s'il est très faible, à condition d'avoir suffisamment de sites.

7.2.3 Robustesse au retatouage

L'expérience reprend le protocole de la section 5.4.3. Nous tatouons tous les documents d'un corpus avec deux clés, l'une après l'autre. Ensuite, nous appliquons l'algorithme de détection avec la première clé.

La figure 7.8 et le tableau 7.8 en montrent le résultat pour une partition en 4 parties, une perte de précision autorisée de 1 mètre et deux clés fixées. Ces résultats montrent que nous pouvons toujours distinguer les documents tatoués et non-tatoués, même après retatouage mais seulement quand les documents ont suffisamment de sommets.

Les tableaux 7.12 de la page 125 donnent le document le plus gros (en nombre de sommets) pour lequel l'expérience a échoué en faisant varier le nombre de parties et la perte de précision autorisée. On constate que le schéma résiste au retatouage pour les documents d'au moins 2000 sommets. Ces tableaux montrent que cette variante du schéma est robuste au retatouage, excepté

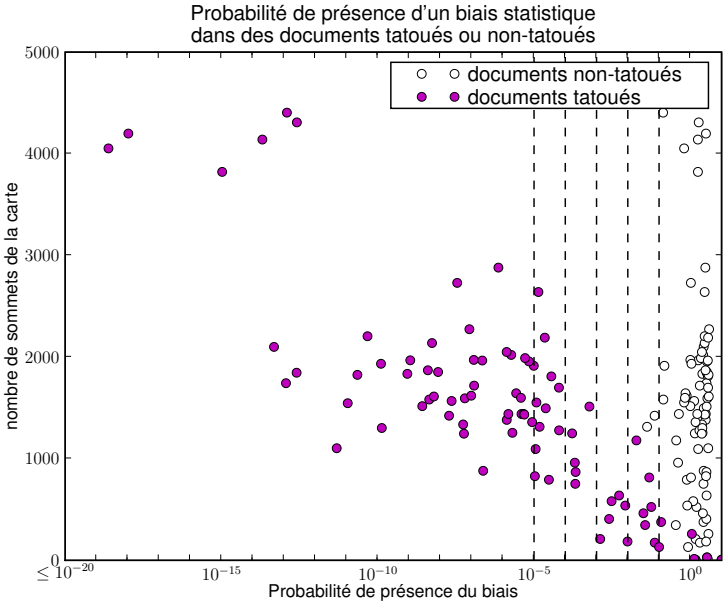


FIG. 7.8 – Résultat de la détection après retatouage du document sur le corpus en ne prenant qu'un seul site par classe de codage.

Seuil	Documents de plus de :					
	0 sommet		200 sommets		400 sommets	
	88 documents		81 documents		77 documents	
	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs	Faux-positifs	Faux-négatifs
10^{-1}	3	7	3	2	3	0
10^{-2}	1	13	1	7	1	4
10^{-3}	0	19	0	12	0	8
10^{-4}	0	24	0	17	0	13
10^{-5}	0	35	0	28	0	24

TAB. 7.9 – Résultat de la détection sur le corpus de test en ne prenant qu'un site par classe de codage.

lorsqu'on tatoue avec un partitionnement en 2 parties. Dans ce cas, le second tatouage va laver les sites impliqués dans le premier tatouage. On voit aussi que le nombre de parties n'influe pas énormément sur la robustesse du schéma face au retatouage. En effet, le nombre de parties donne la proportion de classes de codages qui sont impliqués dans le tatouage. Or, plus le nombre de parties est élevé, plus la proportion de classes de codage impliqués dans le tatouage est faible et plus la quantité de site lavés par la deuxième marque est faible.

Conclusion

Nous avons conçu et testé une variante du schéma pour laquelle on ne choisit qu'un seul site par classe de codage. Cette variante garantit que les tatouages ne sont pas corrélés pour deux clés différentes et on ne risque pas d'attaques basées sur l'étude statistique des grosses classes de codage. Nous avons montré expérimentalement que, comparée au schéma original, cette variante requiert des documents contenant plus de sommets pour être robuste au découpage ou retatouage.

Bien que les résultats soient déjà satisfaisants, on pourraient encore les améliorer (notamment pour le retatouage). Rappelons que nous choisissons aléatoirement le site traité au sein de chaque classe de codage. En décidant d'une heuristique pour choisir le site à traiter au sein de la classe de codage, on éviterait de choisir un site différent lors de la détection que celui choisi pour le marquage. On pourrait alors retrouver la marque dans de plus petits documents.

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	29	29	281	29	347	828	80	155	281	961	815	1968	80	210	1513	1423	2205	6398
10^{-2}	29	80	525	377	407	1517	80	260	1337	1968	1972	2316	155	281	1620	2021	2880	6398
10^{-3}	29	131	793	828	828	1620	80	525	1337	2640	1972	3198	210	793	2138	2205	5126	6398
10^{-4}	29	347	828	869	1972	2021	155	638	1337	2640	2730	5126	281	815	2138	4667	5126	6398
10^{-5}	29	347	1279	1423	1972	2872	174	881	1810	2872	3541	5126	525	881	2274	5126	9544	9544

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	49	103	202	174	558	1496	49	684	1819	998	2638	2443	103	236	2667	2733	2733	3436
10^{-2}	49	199	875	943	947	1819	174	684	1819	2656	2638	3436	236	1196	3109	2733	3436	7814
10^{-3}	174	199	1074	1819	1496	2354	174	892	1819	2688	3208	3436	236	1692	3109	5620	3436	7814
10^{-4}	174	278	1196	1819	2656	2745	477	1196	1819	2733	8122	7814	1235	1692	3109	7727	10921	8122
10^{-5}	199	477	1299	1819	3109	2967	477	1196	3109	3208	8122	8122	1235	2443	7727	8122	10921	10921

TAB. 7.10 – Taille du plus grand document (en nombre de sommets) pour lequel la détection de la marque a échoué (Les corpus issus des mêmes documents sont regroupés et chaque expérience est renouvelée avec 3 clés différentes).

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	1011	1262	2724	2246	2246	3019	1131	1624	2452	3019	4428	4428	1215	2093	3019	4712	3751	4712
10^{-2}	1384	2091	3060	2246	3751	4712	1384	2724	2724	3751	4699	4712	2091	2452	4428	7698	6592	11355
10^{-3}	1384	2091	3060	4712	4428	4712	1737	2724	3019	4428	4699	7698	2093	3060	4428	7698	6592	11355
10^{-4}	1624	2091	3060	4712	4428	4712	1737	3751	4428	4712	6592	7698	2452	4712	4712	7698	6592	12794
10^{-5}	1624	2091	4712	4712	4712	4950	2726	3751	4712	4712	6592	10484	2724	4712	5391	7698	7698	12794

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	706	881	1548	2510	2551	2160	814	1405	2044	2511	2511	2923	1290	1546	2923	2160	2923	2923
10^{-2}	881	1242	2044	2510	2551	3431	1350	2044	2511	2923	2923	3431	2044	2283	3431	3432	2923	2923
10^{-3}	881	1457	2044	2510	3431	3431	1350	2044	2511	3432	3431	3431	2044	2606	3431	3432	3431	3431
10^{-4}	1224	2044	2283	2606	3431	3431	1405	2044	2923	3432	3431	3431	2044	2606	3431	3432	3431	3431
10^{-5}	1224	2044	2606	2606	3431	3431	2044	2511	3431	3432	3431	3431	2044	2606	3431	3432	3431	3431

TAB. 7.11 – Nombre maximum de sommets pour lequel la détection de la marque a échoué après découpage (Les corpus issus du même document original sont fusionnés).

(a) Sur les tronçons de route du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	20013	1255	1255	753	828	828	13552	1246	1846	961	2021	4114	20013	2653	1846	1870	2274	2872
10^{-2}	41202	1255	1957	1434	1853	1990	20013	2640	1990	2640	2205	4114	20013	4082	4667	5126	4232	6398
10^{-3}	41202	1957	2021	2021	2021	2205	41202	4114	4114	2640	2640	4405	20013	5126	4699	9544	5126	7558
10^{-4}	41202	2100	2021	2021	2640	4114	41202	4114	4114	3541	3198	5126	20013	6398	6398	9544	5126	15078
10^{-5}	41202	2653	2730	2653	2640	4114	41202	4114	5126	4763	6398	9544	41202	6398	9017	9544	9544	15078

(b) Sur les limites administratives des communes du Calvados

Perte de précision autorisée (m)	1						3						5					
Nombre de parties	2	4	8	10	12	16	2	4	8	10	12	16	2	4	8	10	12	16
Seuil de détection																		
10^{-1}	10921	875	1142	2688	1198	1074	10096	2656	3208	2638	2656	3436	10921	2688	3436	3436	3109	3436
10^{-2}	10921	2354	1589	2688	1269	1358	10921	2733	3208	3109	8122	8122	10921	2967	3436	3436	3436	10921
10^{-3}	10921	2442	1589	2688	1358	3436	10921	8122	3436	6829	8122	8122	10921	3436	3436	7814	10921	10921
10^{-4}	10921	2656	3208	3027	3208	5620	10921	8122	8122	6829	8122	10921	10921	8122	10921	7814	10921	10921
10^{-5}	10921	2967	3436	3109	7727	5620	10921	8122	8122	7727	8122	10921	10921	8122	10921	10921	10921	10921

TAB. 7.12 – Taille du plus grand document pour lequel la détection de la marque a échoué après ***retatouage*** (Les corpus issus du même document original sont fusionnés, les expériences avec des couples de clés différentes ont été fusionnées).

Conclusion des variantes

Dans ce chapitre, nous avons proposé une variante du schéma pour améliorer la robustesse au découpage et retrouver la marque dans de plus petits documents.

Grâce à une étude détaillée du schéma nous avions vu que les grosses classes de codages peuvent être à l'origine d'une attaque. Afin de palier ce problème, nous avons introduit une variante qui consiste à choisir un site parmi chaque classe de codages. Nous avons montré expérimentalement que cette variante est robuste au découpage et au retatouage mais qu'elle demande de plus gros documents (en nombre de sommets) pour pouvoir détecter la marque, surtout quand le document a été découpé ou retatoué. Nous avons donc le choix entre deux schémas. Si la résistance aux attaques sur les grosses classes de codages n'est pas un critère important pour le tatoueur, il privilégiera la première méthode.

Il existe d'autres façons de se prémunir contre les attaques basées sur l'étude statistique des grosses classes de codage. Nous avions envisagé de masquer la propriété Φ avec une clé mais n'avons pas trouvé de bonne implémentation pour cette idée. Nous pensons cependant qu'il s'agit là d'une solution qu'il serait intéressant de développer.

Notons bien que les deux variantes peuvent être combinées afin d'améliorer la robustesse au découpage et d'éviter les problèmes liés aux grosses classes de codages.

Troisième partie

Généralisation de la méthode

Chapitre 8

Présentation du schéma générique

Sommaire

8.1 Intérêt d'une généralisation

L'état de l'art du tatouage de documents géographiques présenté au chapitre précédent a montré plusieurs schémas de tatouage dérivés de schémas conçus initialement pour d'autres types de données (modèles 3D, dessins vectoriels, etc.). Il est donc possible d'extraire des caractéristiques communes entre plusieurs schémas. Cependant, cette approche a des limites car certains de ces schémas ne tiennent pas toujours compte des spécificités des documents géographiques. Ils peuvent s'intéresser, par exemple, à préserver l'aspect visuel du document mais ne donnent pas de garanties concernant l'aspect qualitatif du document. Or, pour les documents géographiques vectoriels, c'est cet aspect qualitatif qui donne toute sa valeur au document.

Concevoir un schéma de tatouage original demande deux expertises. D'une part, il faut avoir bien compris la problématique du tatouage de documents numériques. D'autre part, il faut une certaine expertise du type de données à tatouer. Concevoir un schéma de tatouage par dérivation d'un schéma existant demande de bien dissocier les parties du premier schéma qui peuvent s'appliquer au second et de celles qui doivent être remplacées. Il faut donc avoir en plus une certaine expertise du type de document tatoué par le schéma initial. Cette façon de

concevoir un schéma de tatouage est donc un travail très difficile car il faut alors réunir les trois expertises.

Pour remédier à ce problème, nous proposons de définir un schéma de tatouage générique. Ce schéma sera défini indépendamment du type de données à tatouer et va synthétiser notre expertise sur le tatouage de données contraintes. Il sera ensuite possible à un expert de concevoir son propre schéma pour le type de document qu'il connaît et suivant les qualités du document qu'il souhaite préserver.

De plus, les schémas implémentés en suivant notre schéma générique formeront une classe sur laquelle nous pourrons travailler. Nous montrerons par exemple que l'on peut définir des protocoles de détection sur le schéma générique qui seront directement applicables pour toutes les implémentations du schéma générique. Par ailleurs, nous pourrons mener une étude directement sur le schéma générique. Par exemple, nous étudierons la robustesse du schéma générique.

8.2 Présentation du schéma générique

Dans cette section, nous présentons notre schéma de tatouage générique. Comme nous le verrons par la suite, ce schéma pourra être implémenté pour différentes classes de documents (données géographiques, bases de données relationnelles, etc.). Le schéma est aveugle, robuste, 0-bit et destiné aux données contraintes. Son objectif principal est de garantir que certaines contraintes du document seront préservées lors du tatouage. C'est la garantie de ces contraintes que nous appellerons préservation de qualité du document.

Pour concevoir ce schéma, nous reprendrons certaines notions comme celles de site et de qualité de document. Nous utiliserons la part d'aléatoire présent dans le site pour insérer la marque. Le codage et la clé décideront où poser cette marque. En partant de la notion de qualité de document et en travaillant localement, nous allons construire un schéma qui garantira que la qualité de document est bien préservée tout au long du processus de tatouage.

8.2.1 Notion de document et de qualité de document

Une fois la classe de documents traités définie, nous définirons la préservation de qualité comme une relation binaire. Notons $\mathcal{D}$ la classe des documents à tatouer. Nous introduisons la relation $Q_{\mathcal{D}}$ pour vérifier si la qualité du document a été préservée d'un document à l'autre. L'ensemble $\mathcal{D}$ et la relation $Q_{\mathcal{D}}$ devront bien sûr être définis pour chaque implémentation de l'algorithme. La relation $Q_{\mathcal{D}}$ est vraie pour deux documents d et d' lorsqu'ils ont la même qualité, on notera alors $Q_{\mathcal{D}}(d, d') =$ vrai ou simplement $Q_{\mathcal{D}}(d, d')$. Notons bien que dans notre définition de $Q_{\mathcal{D}}$, la qualité d'un document se mesure relativement à un document de référence. Pour notre algorithme, nous souhaitons préserver la qualité du document original. En d'autres termes, pour un document $d \in \mathcal{D}$ et sa version tatouée $d_w \in \mathcal{D}$, on souhaite avoir $Q_{\mathcal{D}}(d, d_w)$, quelle que soit la clé de tatouage. Cette relation doit bien sûr être réflexive : il est en effet naturel que la qualité d'un document non transformé soit préservée. Pour les exemples que nous traiterons, la

relation sera symétrique bien que ce ne soit pas nécessaire. Si la transitivité de la relation n'est pas exigée, nous verrons qu'elle est vérifiée pour l'exemple sur les bases de données mais pas sur celui des données géographiques. Nous pouvons maintenant présenter notre algorithme de tatouage.

8.2.2 Présentation de l'algorithme de tatouage

La notion de site est fondamentale dans notre schéma. Grâce à elle, nous pouvons travailler localement sur le document. Nous introduisons $\mathcal{S}$ qui désigne l'ensemble des sites que l'on peut trouver dans la classe des documents $\mathcal{D}$. Nous extrayons les sites du document un par un. Pour chacun d'eux, nous vérifions, à l'aide de la clé, s'il est impliqué dans le marquage du document. Lorsque c'est le cas nous le forçons à satisfaire une certaine propriété pour insérer la marque. Avant de répercuter cette modification au niveau du document, nous nous assurons que cela ne va pas modifier la qualité du document.

Vérifier que la qualité du document a été préservée à chaque modification risque d'être pénalisant en temps de calcul. C'est la raison pour laquelle nous introduisons la relation de qualité de site $Q_{\mathcal{S}}$, qui compare juste 2 sites. Nous montrons dans la partie 8.2.6 que si les fonctions de manipulation des sites sont bien choisies, il suffit de vérifier que la qualité de sites est préservée entre le site extrait et le site modifié pour qu'une répercution de la modification au niveau du document préserve la qualité globale de ce dernier.

L'algorithme 10 présente l'algorithme de tatouage, l'algorithme 11 en est une version plus formelle. Les fonctions X et R permettent de passer du niveau du document au niveau du site et inversement. La fonction X va servir à extraire chaque site d'un document tandis que R va appliquer les modifications du site au niveau du document. Les fonctions C, T_0, M_0, T_1, M_1 serviront à manipuler localement les sites. La fonction C donne le codage d'un site. Les fonctions M_0 et M_1 modifient les sites. Lorsque la relation $Q_{\mathcal{S}}$ est vérifiée entre 2 sites s et s', on peut alors utiliser la fonction R pour remplacer le site s par le site s' dans le document tout en préservant sa qualité.

Dans la suite de cette section, nous détaillerons chaque étape de l'algorithme et donnerons une idée plus précise de ce que nous attendons de chacune de ces fonctions qui doivent être implémentées pour chaque classe de document que l'on souhaite tatouer.

Passage de l'échelle du document à l'échelle du site

La fonction $X : \{1, \dots, m\} \times \mathcal{D} \rightarrow \mathcal{S}$ permet d'extraire chacun des m sites d'un document. $X(i, d)$ va extraire le i-ème site d'un document $d \in \mathcal{D}$ qui contient m sites (avec $i \in \{1, \dots, m\}$). Notons bien que le premier argument de la fonction sert à distinguer chaque site du document et que cette fonction extrait des informations sans modifier le document passé en paramètre.

Algorithm 10: L'algorithme de tatouage commenté

Data: $d \in \mathcal{D}$: le document à tatouer, contenant m sites
Data: p : le nombre de parties de la partition
Data: $k \in \mathcal{K}$: la clé
Result: $w \in \mathcal{D}$: le document tatoué

```
1  begin
2      On créé une copie w de d ;
3      foreach i de 1 à m do
4          Extraction du i-ème site du document w ;
5          Calcul de la partie j à laquelle ce site appartient avec la clé k ;
6          if Le site appartient-il à la première partie et satisfait-il Φ1 ? then
7              On force le site à satisfaire Φ0 ;
8              if Cette modification préserve-t-elle la qualité de site ? then
9                  On répercute les modifications dans le document w ;
10         else if Le site appartient-il à la seconde partie et satisfait-il Φ0 ? then
11             On force le site à satisfaire Φ1 ;
12             if Cette modification préserve-t-elle la qualité de site ? then
13                 On répercute les modifications dans le document w ;
14     return w ;
15 end
```

Algorithm 11: L'algorithme de tatouage

Data: $d \in \mathcal{D}$: le document à tatouer, contenant m sites
Data: p : le nombre de parties de la partition
Data: $k \in \mathcal{K}$: la clé secrète
Result: $w \in \mathcal{D}$: le document tatoué

```
1  begin
2      d_0 ← copie du document d ;
3      foreach i ∈ {1, ..., m} do
4          s_i ← X(i, d_{i-1}) ;
5          j = P_p(C(s_i), k) ;
6          if j = 0 and T_1(s_i) = 1 then
7              s'_i ← M_0(s_i) ;
8              if Q_S(s_i, s'_i) = 1 then d_i ← R(i, d_{i-1}, s') ;
9              else d_i ← d_{i-1} ;
10         else if j = 1 and T_0(s_i) = 1 then
11             s'_i ← M_1(s_i) ;
12             if Q_S(s_i, s'_i) = 1 then d_i ← R(i, d_{i-1}, s') ;
13             else d_i ← d_{i-1} ;
14         else d_i ← d_{i-1} ;
15     w ← d_i ;
16     return w ;
17 end
```

Sélection des sites intervenant dans le tatouage

Nous sélectionnons les sites intervenant dans le tatouage grâce à un codage du site et à la clé secrète. Lorsqu'un site n'intervient pas dans le tatouage, on passe alors au site suivant. Le codage du site est obtenu par la fonction de codage $C : \mathcal{S} \rightarrow \mathbb{N}$ qui associe un code pour chaque site. On souhaite que les codages produits soient les plus variés possibles. On voudrait, dans le meilleur des cas, que chaque codage soit unique. Pour atteindre cet objectif dans l'algorithme de tatouage de données géographiques, nous avions choisi de ne prendre qu'un représentant par classe de codage.

Pour sélectionner les sites, nous les partitionnons en p parties. Les sites des deux premières parties vont porter la marque. Le partitionnement des sites est obtenu par la fonction P_p : $\mathbb{N} \times \mathcal{K} \rightarrow \{0, 1, ..., p-1\}$. La fonction peut être implémentée en utilisant une fonction de hachage classique (md5, sha1). Cette fonction de hachage doit bien disperser les sorties entre les p valeurs comprises entre 0 et $p-1$. Le codage de site et la clé forment les entrées de la fonction de hachage. En sortie, on attend des valeurs entre 0 et $p-1$ telles que :

– lorsque deux sites ont deux codages différents, les parties auxquelles ils appartiennent ne

sont pas corrélées ;
– pour deux clés différentes, les parties associées à un site donné ne sont pas corrélées.

Par conséquent quelqu'un qui connaît la partie associée à un site pour une certaine clé n'aura pas d'information sur la partie associée au même site avec une clé différente. En d'autres termes, si la clé n'est pas connue, il est impossible de déterminer à quelle partie un site sera assigné.

Modification des sites

Nous voulons profiter de l'aspect aléatoire du document pour introduire un biais statistique sur la proportion de sites qui satisfont les propriétés Φ_0 et Φ_1. Pour chaque implémentation du schéma, nous devrons définir ces deux propriétés, qui pour un document non-tatoué, suivent des lois binomiales de paramètres respectifs μ_0 and μ_1. L'objectif du schéma est de forcer un maximum de sites de la première partie de la partition à satisfaire Φ_0 et un maximum de sites de la seconde partie à satisfaire Φ_1. Notons que nous pouvons choisir Φ_1 comme la négation de Φ_0. C'est le choix que nous avons fait pour le schéma de tatouage de données géographiques.

Le biais statistique sera introduit par la modification des sites sélectionnés. Pour vérifier si un site satisfait Φ_0, nous introduisons la fonction $T_0 : \mathcal{S} \rightarrow \{0, 1\}$. Cette fonction doit renvoyer 1 quand le site satisfait Φ_0 et 0 sinon. Pour forcer un site à satisfaire Φ_0, nous introduisons la fonction $M_0 : \mathcal{S} \rightarrow \mathcal{S}$. Nous introduisons aussi les fonctions T_1 et M_1 qui sont à la propriété Φ_1 ce que T_0 et M_0 sont à Φ_0. Nous donnerons plus de détails sur les propriétés Φ_0 et Φ_1 dans la section 8.2.3. nous verrons notamment qu'il est important que les modifications d'un site préservent son codage.

Passage du niveau site au niveau document

Pour répercuter les modifications d'un site au niveau du document sans en modifier la qualité globale, nous introduisons les fonctions $Q_{\mathcal{S}}$ et R. La relation $Q_{\mathcal{S}} : \mathcal{S} \times \mathcal{S}$ permet de vérifier si le remplacement d'un site va préserver la qualité du document. Si tel est le cas, on peut utiliser la fonction $R : \{1, \ldots, m\} \times \mathcal{D} \times \mathcal{S} \rightarrow \mathcal{D}$. $R(i, d, s')$ remplace le i-ème site d'un document $d \in \mathcal{D}$ par le site s' et produit ainsi un nouveau document. On souhaite que le document produit d' vérifie $X(i, d) = s'$.

La section 8.2.6 donnera les conditions que les fonctions X, M, $Q_{\mathcal{S}}$ et R doivent respecter pour s'assurer que l'algorithme de tatouage 11 préserve la qualité du document original.

Bilan

Nous avons introduit l'algorithme de tatouage générique et les fonctions qui le compose : X, C, T_0, T_1, M_0, M_1, $Q_{\mathcal{S}}$ et R. Dans la section suivante nous verrons que ces fonctions sont interdépendantes et nous donnerons les conditions qu'elles doivent remplir pour que l'algorithme de tatouage préserve la qualité du document.

8.2.3 L'introduction du biais statistique

Le marquage du document est obtenu en tirant profit de l'aspect aléatoire des documents à tatouer. Nous introduisons un biais statistique au sein d'un groupe de sites du document. Cette idée est directement reprise de l'algorithme de tatouage de données géographiques.

Manipulation des sites

De façon à manipuler les sites, pour $x \in \{0,1\}$, nous introduisons les fonctions T_x et M_x. Celles-ci sont associées à chaque propriété Φ_x qui suit une loi binomiale de paramètre μ_x. Pour rappel, la fonction T_x sert à tester si un site satisfait Φ_x tandis que M_x sert à modifier un site pour qu'il satisfasse Φ_x.

Pour $x = 0$ et 1, la fonction $T_x : \mathcal{S} \rightarrow \{0,1\}$ retourne 1 lorsque le site satisfait Φ_x, sinon elle vaut 0. Pour un site S pris dans un document non-tatoué, nous aurons $\Pr(T_x(S) = 1) = \mu_x$. On veux que $M_x : \mathcal{S} \rightarrow \mathcal{S}$, la fonction qui essaie de forcer un site à satisfaire Φ_x soit telle que pour un site S choisi aléatoirement sur l'ensemble des sites d'un document non tatoué, $\Pr\big(T_x(M_x(S)) = 1\big) > \mu_x + \varepsilon$ (pour un $\varepsilon > 0$ fixé)

Introduction du biais statistique

Le biais statistique est introduit dans le document lorsqu'un nombre suffisant de sites modifiés ont pu être réintroduits et que les autres sites du document ne sont pas trop modifiés lors de ce remplacement.

Détection du biais statistique

L'algorithme de détection sélectionne les sites qui devraient comporter un biais statistique et utilise la borne de Chernoff pour vérifier sa présence.

Soit un ensemble de n_x sites. En utilisant la fonction T_x nous pouvons compter le nombre m_x de sites qui satisfont Φ_x ($m_x \leq n_x$). La borne de Chernoff donne une borne maximale sur la probabilité d'observer le fait que m_x sites sur n_x satisfont Φ_x sous l'hypothèse que le document n'est pas tatoué.

Soit S la variable aléatoire du nombre de sites satisfaisant la propriété Φ_x. La borne de Chernoff nous donne la borne supérieure de la probabilité que S s'écarte de $n_x\mu_x$ de plus de $|n_x\mu_x - m_x|$:

$$\Pr(|n_x\mu_x - S| \geq |n_x\mu_x - m_x|) \leq 2e^{-2c^2} \quad \text{avec} \quad c^2 = \frac{(n_x\mu_x - m_x)^2}{n_x}$$

Nous introduisons la fonction $Test$ pour calculer la borne de Chernoff et estimer la probabilité d'observer le biais mesuré lorsque le document n'a pas été tatoué.

$$Test(n_x, m_x, \mu_x) = 2e^{-2c^2} \quad \text{avec} \quad c^2 = \frac{(n_x\ \mu_x - m_x)^2}{n_x}$$

Pour pouvoir décider si, oui ou non, un document est tatoué, nous comparons la borne calculée avec un seuil $\lambda \in [0, 1]$. Celui-ci est un paramètre de l'algorithme de détection, son choix est très important. Pour rappel, nous avions testé plusieurs valeurs de seuil pour les données géographiques, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4} et 10^{-5}. Plus le seuil est élevé, plus la chance de détecter un faux positif augmente. Inversement, si il est trop faible, l'algorithme risque de manquer la détection d'une marque au sein d'un document, il est donc essentiel de bien régler ce seuil.

L'algorithme de détection

L'algorithme 12 utilise les fonctions X, C, T_0 et T_1 pour détecter la présence de la marque au sein d'un document. L'algorithme extrait les sites d'un document. Pour chaque site, il vérifie, avec la clé, à quelle partie appartient le site et s'il vérifie Φ_0 ou Φ_1. Cela permet ensuite de tester si un biais statistique sur la proportion de sites vérifiant Φ_0 et Φ_1 est présent au sein des deux premières parties de la partition. Lorsque la probabilité d'avoir une proportion de sites satisfaisant les propriétés Φ_0 et Φ_1 dans les deux premières parties de la partition est inférieure au seuil λ, nous considérons que le document est tatoué. En d'autres termes, si le biais mesuré est significatif, nous concluons que le document est tatoué.

Notons que la boucle principale de l'algorithme peut être parallélisée pour rendre l'algorithme plus rapide.

8.2.4 Identification du propriétaire

L'identification du propriétaire est obtenue grâce à la clé, chaque clé permettant de créer une partition différente de l'ensemble des sites d'un document. Tant que les codages des sites n'ont pas été modifiés par des transformations, le partitionnement des sites effectué lors du tatouage est reproductible par l'algorithme de détection (à condition de connaître la clé). D'autre part, celui qui ne connaît pas la clé ne pourra pas reproduire ce partitionnement. Le propriétaire du document utilisera la clé pour montrer la présence du biais statistique et prouver que le document lui a été volé.

Le partitionnement ne doit pas être modifié lors du tatouage du document il faut donc que la modification des sites ne modifient pas les codages associés aux sites. En d'autres termes, il faut montrer que les implémentations choisies pour les différentes fonctions de l'algorithme 11 préservent bien le codage des sites lors du tatouage.

8.2.5 Invisibilité de la marque

Pour que la marque ne soit pas visible au niveau du document, il faut conserver les distributions de Φ_0 et Φ_1 au niveau global du document. Nous devons donc nous assurer que l'augmentation de la proportion de sites qui satisfont Φ_0 dans la première partie de la partition sera compensée par la diminution de la proportion de ceux qui satisfont Φ_0 dans la seconde

Algorithm 12: L'algorithme de détection

Data: $d \in \mathcal{D}$: le document à tatouer, contenant m sites
Data: $k \in \mathcal{K}$: la clé
Data: μ_0, μ_1 : la probabilité qu'un site satisfasse respectivement Φ_0 et Φ_1
Data: λ : le seuil de détection
Result: *True* : si d est tatoué avec k, sinon *False*
begin
  $n_0 \leftarrow 0$; $m_0 \leftarrow 0$; $n_1 \leftarrow 0$; $m_1 \leftarrow 0$;
  foreach $i \in \{1, \dots, m\}$ **do**
    $s \leftarrow X(i, d)$;
    $c \leftarrow C(s)$;
    $j = P_p(c, k)$;
    if $j = 0$ **then**
      $n_0 \leftarrow n_0 + 1$;
      if $T_0(s) = 1$ **then** $m_0 \leftarrow m_0 + 1$;
    else if $j = 1$ **then**
      $n_1 \leftarrow n_1 + 1$;
      if $T_1(s) = 1$ **then** $m_1 \leftarrow m_1 + 1$;
  return $Test(n_0, m_0, \mu_0)\ Test(n_1, m_1, \mu_1) < \lambda$;
end

partie. Il faut bien entendu que les modifications de sites satisfaisant Φ_1 soient aussi compensées d'une partie à l'autre.

8.2.6 Preuve de préservation de qualité du document

Dans cette section, nous montrons que la qualité du document original est préservée par l'algorithme 11 sous certaines conditions que nous précisons. Nous reprenons les notations de cet algorithme.

Nous allons donner les conditions suffisantes sur les fonctions X, M, $Q_{\mathcal{S}}$ et R pour que la qualité du document original $d = d_0$ contenant m sites soit préservée pour les documents $(d_i)_{i \in \{1,\dots,m\}}$ produits à chaque tour de boucle, et qui finalement aboutissent au document tatoué w, qui vérifie $Q_{\mathcal{D}}(d, w)$.

Théorème 1 *Supposons que pour tout document $d \in \mathcal{D}$ contenant m sites, les fonctions X, M, $Q_{\mathcal{S}}$ et R respectent la formule :*

$$\forall i \in \{1, \dots, m\} \Big(Q_{\mathcal{D}}(d, d_{i-1}) \wedge Q_{\mathcal{S}}(s_i, s'_i) \Big) \implies Q_{\mathcal{D}}(d, d_i)$$
$$avec \begin{cases} s_i = X(i, d_{i-1}) \\ s'_i = M(s_i) \\ d_i = R(i, d_{i-1}, s'_i) \end{cases} \tag{8.1}$$

Alors, le document w produit par l'algorithme 11 à partir de d vérifie $Q_{\mathcal{D}}(d, w)$.

Pour démontrer le théorème, nous commencerons par montrer que lorsque la formule 8.1 est vraie, dans l'algorithme 11, on a bien $Q_{\mathcal{D}}(d_1, d_{i+1})$ quand $Q_{\mathcal{D}}(d_1, d_i)$. Nous utiliserons ensuite ce résultat pour montrer que, sous l'hypothèse que la formule 8.1 est vraie, pour tout document de départ $d_0 \in \mathcal{D}$ contenant m sites, l'algorithme 11 produit une suite de documents $(d_i)_{i\in\{1,\dots,m\}}$ pour lesquels on a $Q_{\mathcal{D}}(d_0, d_i) = 1$. Ce résultat nous permettra de terminer la démonstration du théorème 1.

Montrons que si la formule 8.1 est vraie alors, à la fin de la i-ème itération de l'algorithme 11, quand $Q_{\mathcal{D}}(d_0, d_{i-1})$, on a $Q_{\mathcal{D}}(d_0, d_i)$. Pour arriver à ce résultat, il suffit de considérer tous les chemins possibles dans la boucle principale de l'algorithme. On reprendra ici les notations de l'algorithme 11 :

- lorsque le site extrait n'intervient pas dans le marquage, alors $d_i = d_{i-1}$ et on a, par hypothèse, $Q_{\mathcal{D}}(d_0, d_{i-1})$. Par conséquent, on a bien $Q_{\mathcal{D}}(d_0, d_i)$;
- lorsque le site extrait intervient dans le tatouage et que $\neg Q_{\mathcal{S}}(s, s')$, pour les mêmes raisons, on a bien $Q_{\mathcal{D}}(d_0, d_i)$;
- enfin, nous avons posé l'hypothèse que X, M, $Q_{\mathcal{S}}$ et R respectent bien la relation 8.1. Par conséquent, lorsque le site extrait intervient dans le marquage et que $Q_{\mathcal{S}}(s, s')$, alors on a bien $Q_{\mathcal{D}}(d_0, d_i)$.

Montrons que si la formule 8.1 est vraie, alors pour tous les documents $(d_i)_{i\in\{1,\dots,m\}}$ produits par l'algorithme 11, on a bien $Q_{\mathcal{D}}(d_0, d_i)$. Ce résultat s'obtient simplement par récurrence :

- par définition, la relation $Q_{\mathcal{D}}$ est réflexive et nous avons $Q_{\mathcal{D}}(d_0, d_0)$;
- supposons que $Q_{\mathcal{D}}(d_0, d_{i-1})$ est vraie. Par hypothèse, X, M, $Q_{\mathcal{S}}$ et R respectent la relation 8.1. Comme $Q_{\mathcal{D}}(d_0, d_i)$ est vraie, d'après le résultat précédent, on a donc bien $Q_{\mathcal{D}}(d_0, d_i)$.

Terminons la démonstration du théorème 1. Dans l'algorithme 11, le document original et son tatoué sont respectivement les documents d_0 et d_m. D'après le résultat précédent, on a $Q_{\mathcal{D}}(d_0, d_m)$. Or on a $d_m = w$. Cela établit le théorème 1.

Conclusion

Nous avons présenté les algorithmes de tatouage et de détection génériques. Nous avons vu que ces schémas reposent sur la notion de site et qu'ils sont composés de fonctions pour manipuler les sites. Ces fonctions seront implémentées spécifiquement suivant la classe de document à tatouer et les qualités que l'ont souhaite préserver sur les documents.

Nous verrons deux exemples d'applications du schéma dans la section suivante. Cela nous permettra de mieux imaginer comment l'appliquer à d'autres types de documents. Intuitivement, retenons que le codage d'un site doit s'opérer sur les informations importantes du site. On peut

se baser sur les qualités de sites à préserver pour définir le codage. Par contre, les propriétés Φ_0 et Φ_1 doivent être définies sur les informations du sites assez lâches pour qu'on puisse forcer les sites à les satisfaire tout en préservant la qualité de site et le codage. Il faut donc que le codage et les propriétés soient définis sur des espaces les plus indépendants possibles.

Nous avons vu qu'un avantage de notre schéma était de rendre cette préservation de qualité au niveau du site suffisante pour préserver la qualité au niveau du document.

Chapitre 9

Application du schéma générique

Sommaire

Ce chapitre présente deux exemples d'application du schéma générique pour deux types de données différentes. Nous donnerons tout d'abord une implémentation d'un schéma pour les données géographiques qui est très proche de celui présenté dans la partie II. Nous présenterons ensuite une implémentation pour le tatouage de bases de données. Ces deux exemples pourront

donner des intuitions au lecteur pour appliquer le schéma sur de nouveaux types de données ou pour d'autres qualités à préserver.

9.1 Application au tatouage de données géographiques

Cette section commencera par préciser la classe de documents qui nous intéresse et quelle qualité du document nous souhaitons préserver lors du tatouage. De là, nous donnerons la définition du site et de la qualité de site. Nous donnerons ensuite les implémentations choisies pour le codage et les propriétés Φ_0 et Φ_1.

9.1.1 Documents $\mathcal{D}$ et préservation de qualité du document $Q_{\mathcal{D}}$

Dans le chapitre 4, nous avions déjà présenté les données géographiques considérées (section 4.1.1, page 42) ainsi que la notion de qualité globale à préserver (section 4.1.2, page 42) aux niveaux topologiques et métriques. Pour rappel, nous travaillons sur des données géographiques vectorielles et nous considérons que le schéma préserve la qualité du document si et seulement si l'algorithme de tatouage :

- préserve la triangulation de Delaunay associée aux sommets du documents ;
- borne le déplacement des sommets par la perte de précision autorisée (donnée par l'utilisateur).

Notons que les contraintes fortes de cet exemple sont d'ordre topologique. Nous avons donc tout intérêt à définir la qualité de site et le codage sur cet espace. Nous avons certaines libertés sur les contraintes métriques, nous utiliserons donc cet espace pour les propriétés Φ_0 et Φ_1.

9.1.2 Site $\mathcal{S}$ et préservation de qualité de site $Q_{\mathcal{S}}$

Nous reprenons les notions de site et de préservation de qualité de site détaillées respectivement dans les sections 4.3.1 (page 51) et 4.3.2 (page 52).

Rappelons rapidement qu'un site est toujours composé :

- d'un sommet central (noté c) ;
- de ses voisins dans la triangulations (notés n_i) ;
- des arêtes entre tous ces sommets (notées E_c) ;
- des sommets opposés au point central sur les faces miroirs à ses faces adjacentes (ces sommets sont notés m_i).

D'autre part, la qualité de site est préservée lorsque toutes les conditions suivantes sont vérifiées :

- seules les positions des sommets centraux des deux sites différent ;
- la distance entre les sommets centraux des deux sites comparés est inférieure à la perte de précision autorisée ;
- les connexions entre les sommets ne sont pas modifiées ;

– la nouvelle position du sommet central respecte les contraintes sur les cercles circonscrits détaillées dans la section 4.3.2 (page 52).

9.1.3 Fonction d'extraction des sites X

La fonction d'extraction d'un site est simple lorsqu'on a calculé la triangulation de Delaunay des sommets composant le document. Il suffit d'ordonner les sommets suivant un ordre quelconque. La fonction $X(i, d)$ extrait le site dont le sommet central est le i-ème sommet du document d suivant cet ordre. Notons que, suivant l'ordre retenu, le tatouage obtenu peut être différent.

9.1.4 Fonction de remplacement R

Nous utiliserons la fonction de remplacement uniquement lorsque la qualité du site est préservée. Dans ce cas, seule la position du sommet central aura changé. L'implémentation de $R(i, d, s)$ va consister à placer le i-ème sommet du document d à la position du sommet central du site s. Ici, le i-ème sommet doit être pris suivant le même ordre que celui qui a été retenu pour la fonction d'extraction.

9.1.5 Fonction de codage des sites C

La fonction de codage reprend la définition de la section 4.4.2 (page 58). Elle se base sur les connexions entre les sommets du site. Celles-ci sont préservées tant que le voisinage du sommet central ne change pas, ce qui est le cas lorsque la triangulation de Delaunay est préservée.

9.1.6 Propriétés Φ_0 et Φ_1

Pour les propriétés Φ_0 et Φ_1, nous reprenons les idées énoncées dans la section 4.4.4 (page 62). Un site satisfait la propriété Φ_0 lorsque la distance discrète entre son sommet central et le barycentre de ses voisins est paire. La propriété Φ_1 sera la négation de Φ_0. Ces propriétés sont relativement robustes tant que les sommets sont peu déplacés.

9.1.7 Preuve de préservation de qualité

On veut montrer que la relation 8.1 est bien vérifiée. La qualité de document est définie de façon à la fois topologique et métrique. Pour vérifier que la qualité du document est bien préservée nous procéderons en deux étapes. Nous montrerons d'abord que la qualité topologique est bien préservée et ensuite que la qualité métrique l'est aussi.

Préservation de la topologie À la i-ème étape, on modifie au plus un sommet du document. Lorsque la préservation de la qualité de site est vérifiée, on respecte par définition les contraintes sur les cercles circonscrits. Donc, aucun triangle n'a pu disparaître ni apparaître au passage de d_i à d_{i+1}, ce qui préserve la triangulation.

Supposons maintenant que d_i ait la même triangulation que d_0. On vient de voir que lorsque la qualité de site est préservée, d_{i+1} a la même triangulation que d_i. Par transitivité, d_{i+1} a donc bien la même triangulation que d_0.

Préservation de la métrique Dans l'algorithme 11, le document d_i est obtenu par des transformations successives de d_o qui traitent les i premiers sites du document. Les sites sont traités séquentiellement, le traitement du site suivant va donc impliquer un site dont le sommet central n'a pas encore été déplacé. De plus, si la qualité de site est préservée, ce déplacement reste dans la limite de perte de précision autorisée.

Quand le document d_i a la même qualité que d_0, les sommets déplacés restent dans la perte de précision autorisée. On vient de voir que, si la qualité de site est préservée, on déplace un sommet non encore traité dans les limites autorisées. On va donc produire un document d_{i+1} pour lequel tous les sites déplacés resteront dans les limites de perte de précision autorisée.

Conclusion de la preuve de préservation de qualité Nous avons montré que lorsque nous avons un document d_{i-1} de même qualité que d_0 et que l'on remplace un site par un autre de même qualité, le document d_i préserve la qualité de d_0. Le théorème 1 peut donc s'appliquer, on a donc la garantie que la qualité du document original est préservée par l'algorithme de tatouage.

9.1.8 Préservation des codages

Nous venons de montrer que la triangulation du document original est préservée lors du tatouage. Or, avec la même triangulation, le voisinage des sommets centraux de tous les sites du document est conservé. Nous avons donc bien préservé les codages des sites. Nous pouvons ensuite reproduire le partitionnement et retrouver le biais statistique.

Conclusion sur le tatouage de documents géographiques vectoriels

L'algorithme présenté dans ce chapitre montre bien comment le schéma présenté au chapitre 2 est une implémentation particulière du schéma générique. Dans la partie suivante, nous verrons que le schéma générique peut s'appliquer à un autre type de documents.

9.2 Application au tatouage de base de données

Nous souhaitons maintenant tatouer une base de données relationnelle. Nous reprenons le problème énoncé par D. Gross-Amblard dans [Gross-Amblard, 2003] que nous avons résumé dans la section 2.3 (page 19). Ce schéma consiste à tatouer une base de données tout en préservant une requête de somme sur un attribut d'un ensemble d'enregistrements de la base. Le problème consiste à modifier les valeurs intervenant dans la somme tout en préservant la somme totale. De plus, la modification de chaque valeur sera bornée.

Imaginons par exemple que nous disposons d'une base de données de productions d'usines. Ce schéma permet d'insérer une marque dans le document tout en préservant la somme totale des productions et en bornant les modifications de la production de chaque usine. Pour des raisons de cohérence de la base, on s'interdira aussi de modifier la valeur de la clé primaire ou de l'identifiant de chaque enregistrement.

Cette section va commencer par modéliser le problème et présenter les principes de base pour le résoudre. Nous présentons ensuite le schéma proprement dit avant de passer aux expérimentations.

9.2.1 Modélisation du problème

Pour simplifier le problème, nous considérons qu'un enregistrement est un couple (i, b) où i est un identifiant robuste de l'enregistrement et b est un bit modifiable (typiquement un bit de poids faible) de l'attribut que l'on veut sommer. Nous supposons que b vaut 1 avec une probabilité μ. La méthode que nous présentons ici peut facilement être étendue lorsque l'on dispose de plusieurs bits à modifier.

9.2.2 Principes du schéma

Pour garantir que la somme totale n'est pas affectée par la modification des enregistrements, nous traiterons les enregistrements par couples. Ainsi, la modification d'un bit modifiable du couple sera compensée par la modification de l'autre bit. Il est clair qu'en préservant la somme des bits modifiables, nous préserverons la somme totale de tous les bits. Cette idée de compensation est reprise de Gross-Amblard [Gross-Amblard, 2003].

Contrairement au schéma original, nous ne souhaitons pas que notre schéma utilise un ordre sur les enregistrements. De ce fait, la suppression ou l'insertion d'enregistrements dans la base de données ne perturbera pas trop la détection.

Nous allons créer autant de sites que de couples d'enregistrements. Cela va poser un problème de complexité car l'algorithme sera quadratique en le nombre d'enregistrements de la base. Nous verrons que l'on peut résoudre ce problème en regroupant les enregistrements dans des groupes relativement petits que l'on traitera comme autant de bases indépendantes. Par ailleurs, plus le nombre de sites augmente, plus le marquage de nouveaux sites risque de laver les sites précédemment marqués. Nous verrons que ce problème se résout simplement en choisissant bien le nombre de parties de la partition afin de ne pas modifier trop de sites.

9.2.3 Documents considérés $\mathcal{D}$ et qualité à préserver $Q_{\mathcal{D}}$

Comme nous l'avons vu précédemment, un document est vu comme un ensemble de couples (i, b). Le bit modifiable b vaut 1 avec une probabilité μ.

On dira que la qualité de la base de données est préservée entre une base d et cette base tatoué d_w lorsque :

- les identifiants des enregistrements sont les mêmes dans d et d_w ;
- la somme des bits modifiables des enregistrements est identique dans d et d_w.

Si ces deux conditions ne sont pas vérifiées, la qualité de la base de données ne sera pas préservée.

9.2.4 Notion de site $\mathcal{S}$ et de qualité de site $Q_{\mathcal{S}}$

Nous définissons le site comme un couple d'enregistrements ordonnés $\big((i_1, b_1), (i_2, b_2)\big)$. Le schéma va donc créer autant de sites que de couples d'enregistrements ordonnés de la base. La qualité de deux sites est préservée lorsque les identifiants et la somme des deux bits modifiables restent inchangés.

Nous produisons donc $n\ (n-1)$ sites pour une base contenant n enregistrements. Sous réserve que le traitement d'un site soit en temps constant, la complexité temporelle du tatouage et de la détection sera donc quadratique en le nombre d'enregistrements. À la fin de cette section, nous présenterons une variante pour avoir une complexité linéaire en le nombre d'enregistrements.

9.2.5 Extraction des sites X et remplacement R

L'extraction des sites est très simple. On itère sur tous les couples d'enregistrements. Notons bien que pour chaque paire d'enregistrements, nous produirons 2 sites $((i_1, b_1), (i_2, b_2))$ et $((i_2, b_2), (i_1, b_1))$.

La fonction de remplacement est très simple elle aussi, elle applique la valeur du bit b_1 (resp. b_2) à l'enregistrement d'identifiant i_1 (resp. i_2).

9.2.6 Codage des sites C

Les identifiants des enregistrements composant le site forment un excellent espace pour coder le site. En effet, lorsque tous les identifiants sont différents, le codage de chaque site sera unique. De plus, les identifiants doivent rester inchangés lors du tatouage. Les codages des sites ne seront donc pas affectés par le marquage et on pourra reproduire le partitionnement lors de la détection.

Le couple formé par les identifiants des deux enregistrements qui composent le site rerésente le codage du site.

9.2.7 Propriétés Φ_0 et Φ_1

Un site $s = \big((i_1, b_1), (i_2, b_2)\big)$ satisfait la propriété :

- Φ_0 lorsque $b_1 = 0$ et $b_2 = 1$;
- Φ_1 lorsque $b_1 = 1$ et $b_2 = 0$.

Nous avions posé l'hypothèse qu'une proportion μ des bits modifiables valent 1. Nous avons donc la même proportion $\mu\ (1 - \mu)$ de sites qui satisfont soit Φ_0, soit Φ_1.

Par exemple, lorsqu'on fixe $\mu = 50\%$ on a 25% des sites qui satisfont Φ_0 et 25% qui satisfont Φ_1. Notons que plus μ est proche de 50%, plus la proportion de sites impliqués dans le tatouage sera élevée.

La définition des fonctions T_0, T_1 est triviale, elle consiste à vérifier la valeur des bits modifiables du site. Les fonctions M_0 et M_1 consistent simplement à forcer b_1 et b_2 aux valeurs adéquates.

9.2.8 Nombre de parties de la partition p

Comme nous travaillons sur tous les couples d'enregistrements possibles, nous aurons beaucoup de collisions, c'est-à-dire de sites qui ont en commun au moins un enregistrement. La modification d'un site va donc en perturber beaucoup d'autres. Afin de limiter ces perturbations, nous allons limiter le nombre de sites impliqués dans le marquage en jouant sur p, le nombre de parties de la partition. En prenant $p = n$, le nombre d'enregistrements, nous limiterons les perturbations en ne modifiant que de l'ordre de n sites. Comme nous le verrons dans les expérimentations, avec une partition en $p = n$ parties nous pouvons tatouer des bases contenant au moins 200 enregistrements.

9.2.9 Preuve de préservation de qualité

On veut montrer que la relation 8.1 est bien vérifiée.

À la i-ème étape, par définiton, lorsque la qualité de site est respectée, la somme des bits modifiables du site est inchangée. De plus, lors du remplacement on ne modifie que les deux enregistrements concernés, donc les autres bits de la base restent inchangés. Par conséquent, la somme totale des bits modifiables de la base d_{i+1} est identique à celle de d_i.

Supposons que la base d_i ait la même somme que d_0. Nous venons de montrer que lorsque la qualité de site est préservée la somme des bits modifiables l'est aussi de d_i à d_{i+1}. Dans ce cas, la qualité de la base sera bien préservée entre d_0 et d_{i+1}.

La relation 8.1 est donc vérifiée. Par conséquent, ce schéma préserve bien la qualité du document original.

9.2.10 Passage en complexité linéaire

Nous avons un algorithme quadratique en le nombre d'enregistrements traités. Voyons maintenant comment adapter l'algorithme pour atteindre une complexité linéaire. L'astuce consiste à distribuer l'ensemble des n enregistrements de la base dans des groupes de taille maximale q fixée. Le groupe auquel appartient un site sera fixé en fonction de son identifiant. Nous traiterons chacun des $\lceil \frac{n}{q} \rceil$ groupes indépendamment des autres comme si chaque groupe appartenait à une base de données indépendante. Le nombre de sites de chaque groupe est au plus q $(q-1)$. Le temps de traitement total est donc de l'ordre de $\lceil \frac{n}{q} \rceil$ q $(q-1)$. Comme q est une constante, la complexité en temps de tatouage ou détection pour une base de données de n enregistrements est donc $O(n)$.

On peut implémenter le découpage en paquets en utilisant une fonction de hachage $H_k()$ qui donne un haché de l'entrée entre 0 et $k-1$. On souhaite former $k = \lceil \frac{n}{q} \rceil$ paquets, cha-

cun contenant environ q enregistrements. Les paquets sont numérotés de 0 à $k-1$. On place l'enregistrement d'identifiant i dans le paquet numéroté $H_k(i)$.

Le groupement des enregistrements peut être effectué avant d'itérer sur les sites, il s'agit alors d'un prétraitement. Cependant, on peut aussi itérer sur les sites du document en même temps que l'on effectue le groupement des enregistrements. L'algorithme est simple ; au fur et à mesure que l'on ajoute un enregistrement dans un des groupes, on utilise les enregistrements déjà présents dans le groupe pour construire les sites dans lesquels apparait l'enregistrement traité.

9.2.11 Validation expérimentale

Nous validons notre schéma sur des bases de données aléatoires construites suivant notre modèle en fixant $\mu = 50\%$. Nous prenons des corpus de bases de données de tailles différentes sur lesquelles nous appliquons l'algorithme de détection avant et après tatouage. Nous avons construit le tableau 9.1 qui synthétise le nombre de faux-positifs et de faux-négatifs lorsqu'on renouvelle l'expérience 100 fois en prenant $p = n$. Notons que pour les expériences présentées dans le tableau, comme les bases restent petites, nous n'avons pas eu besoin d'appliquer le passage en complexité linéaire.

Le tableau montre que le tatouage de base de données fonctionne. Pour les bases contenant au minimum 400 enregistrements, nous distinguons parfaitement les documents tatoués et non-tatoués. De plus, le nombre de faux positifs pour les bases de 300 enregistrements est très faible.

Les tableaux 9.2 et 9.3 illustrent l'évolution des résultats du schéma lorsqu'on fixe $p = 2n$ et $p = 4n$. Ces deux tableaux montrent que les résultats sont légèrement meilleurs pour $p = 2n$ et beaucoup moins bon lorsque $p = 4n$. Pour une base de données de 400 enregistrements, il faudra donc prendre p entre n et $4n$.

Seuil de détection	Taille de la base de données :							
	100 enregistrements		200 enregistrements		300 enregistrements		400 enregistrements	
	$\sim$ 10000 sites		$\sim$ 40000 sites		$\sim$ 90000 sites		$\sim$ 160000 sites	
	100 expériences		100 expériences		100 expériences		100 expériences	
	FP	FN	FP	FN	FP	FN	FP	FN
10^{-1}	1	25	0	2	1	0	3	0
10^{-2}	0	66	0	12	0	2	0	0
10^{-3}	0	84	0	41	0	3	0	0
10^{-4}	0	97	0	65	0	15	0	1
10^{-5}	0	99	0	80	0	39	0	10

TAB. 9.1 – Résultat du schéma du tatouage de base de données avec $p = n$. Les colonnes nommées FP et FN donnent respectivement le nombre de faux positifs et de faux négatifs.

Seuil de détection	Taille de la base de données :							
	100 enregistrements		200 enregistrements		300 enregistrements		400 enregistrements	
	$\sim$ 10000 sites		$\sim$ 40000 sites		$\sim$ 90000 sites		$\sim$ 160000 sites	
	100 expériences		100 expériences		100 expériences		100 expériences	
	FP	FN	FP	FN	FP	FN	FP	FN
10^{-1}	0	27	0	1	0	0	0	0
10^{-2}	0	57	0	10	0	1	0	0
10^{-3}	0	79	0	29	0	3	0	0
10^{-4}	0	92	0	52	0	13	0	1
10^{-5}	0	95	0	77	0	26	0	2

TAB. 9.2 – Résultat du schéma du tatouage de base de données avec $p = 2n$. Les colonnes nommées FP et FN donnent respectivement le nombre de faux positifs et de faux négatifs.

Seuil de détection	Taille de la base de données :							
	100 enregistrements		200 enregistrements		300 enregistrements		400 enregistrements	
	$\sim$ 10000 sites		$\sim$ 40000 sites		$\sim$ 90000 sites		$\sim$ 160000 sites	
	100 expériences		100 expériences		100 expériences		100 expériences	
	FP	FN	FP	FN	FP	FN	FP	FN
10^{-1}	0	29	0	6	1	0	0	0
10^{-2}	0	67	0	26	0	4	0	1
10^{-3}	0	87	0	50	0	16	0	1
10^{-4}	0	93	0	74	0	41	0	9
10^{-5}	0	98	0	87	0	61	0	18

TAB. 9.3 – Résultat du schéma du tatouage de base de données avec $p = 4n$. Les colonnes nommées FP et FN donnent respectivement le nombre de faux positifs et de faux négatifs.

9.2.12 Comparaison avec la méthode originale

Pour montrer les apports de notre méthode, nous comparons ici notre schéma avec celui de Gross-Amblard et Lafaye.

Dans le schéma original, les auteurs commence par trier des enregistrements en fonction d'une clé. Cette étape préliminaire est réalisée en temps $O(n(lg(n))$. Nous avons vu qu'en implémentant correctement notre schéma, nous n'avons pas besoin de prétraitement. Nous pouvons donc détecter la marque dans une base au fur et à mesure du parcours des enregistrements. D'autre part si l'algorithme de détection s'arrête dès que le seuil de détection est dépassé, il n'a pas à parcourir la base entièrement.

De plus, nous pouvons tatouer la base de façon incrémentale, Pour chaque nouvel enregistrement inséré, il suffit de générer les nouveaux sites et de les traiter suivant la boucle de l'algorithme 11.

Le schéma de Gross-Amblard itère deux à deux les enregistrements de la liste triée. La suppression et l'insertion d'un seul enregistrement, surtout si elle a lieu au début de la liste triée peut perturber la synchronisation. Lorsque c'est le cas, le couplage d'enregistrements est complètement différent, ce qui va empêcher l'algorithme de détection de retrouver la marque. Notre schéma est par contre peu sensible à l'ajout ou la suppression de quelques enregistrements, ce qui s'explique par le fait que nous n'avons pas besoin d'une notion d'ordre.

Pour être réellement convaincus de l'efficacité de la méthode, il reste à mener des tests comparables à ceux effectués par Gross-Amblard. Nous devrons par exemple valider notre schéma sur des bases de données réelles et valider sa robustesse face à certaines transformations ce qui demande beaucoup de temps.

Conclusion

Nous avons présenté des implémentations du schéma générique pour deux types de données : les données géographiques et les bases de données. Ces implémentations doivent être vues comme des exemples qui donneront des intuitions au lecteur pour appliquer le schéma générique à de nouveaux types de données ou pour d'autres contraintes intervenant pour la préservation de qualité de document.

Chapitre 10

Étude du schéma générique

Sommaire

Dans ce chapitre, nous allons étudier l'impact des transformations sur le schéma générique. Comme nous l'avons vu dans la section précédente, l'algorithme de tatouage consiste à introduire un biais statistique dans un sous-ensemble des sites d'un document. L'algorithme de détection quantifie ce biais et un seuil permet de décider si le document est tatoué ou non. Une transformation du document va diminuer la présence du biais. Grâce à la généralisation de la méthode, nous pouvons étudier indépendamment, d'une part, l'influence d'une transformation sur l'ensemble des sites et, d'autre part, l'influence de la modification de l'ensemble des sites sur la présence du biais.

Plus précisément, une transformation du document se traduit par une combinaison de suppressions, d'ajouts et de modifications de sites. En pratique, cette correspondance n'est pas toujours simple à quantifier car elle dépend fortement du type de document et de la transformation que l'on considère. On peut cependant l'estimer expérimentalement ou proposer une modélisation selon le type de document.

Dans cette section, nous partons du principe que nous connaissons la correspondance entre une transformation du document et une modification de l'ensemble des sites. Il nous reste alors à quantifier la diminution du biais statistique en fonction des modifications de l'ensemble de site. Cette étude peut être menée sans connaissance à priori du type de document traité. Nous nous intéressons à trois types de modifications fondamentales : la suppression d'un sous-ensemble de sites, l'ajout de nouveaux sites non biaisés et la modification d'un sous-ensemble de sites.

Bien que ce ne soit pas forcément le cas dans la réalité, nous supposons dans cette étude que les sites supprimés ou modifiés sont choisis aléatoirement sur l'ensemble des sites du document.

10.1 Prérequis

Rappelons que pour évaluer la présence de la marque, nous utilisons la fonction $Test$ qui donne une borne sur la probabilité d'observer le biais mesuré lorsque le document n'a pas été tatoué. Cette fonction utilise la borne de Chernoff.

Rappelons que pour $x \in \{0,1\}$, sur l'ensemble de sites issu d'un document non-tatoué, Φ_x suit une loi binomiale de paramètre μ_x. Nous notons n_x le nombre de sites de la partie x de la partition. La variable m_x représente le nombre de sites de cette partie qui satisfont Φ_x. La proportion de sites qui satisfont Φ_x est notée $\beta_x = \frac{m_x}{n_x}$. En reprenant ces notations, on peut écrire la fonction $Test$ sous la forme :

$$Test(n_x, n_x\beta_x, \mu_x) \quad = \quad 2e^{-2n_x(\mu_x-\beta_x)^2}$$

Pour les différentes modifications de l'ensemble des sites proposées, nous faisons l'hypothèse que les sites sont répartis dans les différentes parties de la partition en suivant une distribution uniforme. Les parties de la partition sont donc toutes à peu près de la même taille.

10.2 Suppression de sites

Nous étudions la suppression d'un sous-ensemble de sites choisis aléatoirement en suivant une loi uniforme parmi un ensemble de n sites de départ. On note $\nu \in [0,1]$ la proportion de sites supprimés de l'ensemble des sites de départ.

Si les sites supprimés sont choisis aléatoirement, nous pouvons considérer que supprimer une proportion ν de l'ensemble des sites total revient à supprimer une proportion ν de sites dans chaque partie. La quantité de sites dans la partie x du document produit devient $(1-\nu)n_x$ et la quantité de site vérifiant Φ_x dans cette partie devient $(1-\nu)\beta_x n_x$. On obtient donc la borne de Chernoff suivante :

$$Test((1-\nu)n_x, (1-\nu)n_x\beta_x, \mu_x) \quad = \quad 2e^{-2(1-\nu)n_x(\mu_x-\beta_x)^2}$$

L'intensité de la marque sera donc la même que si l'on avait tatoué un document contenant $n\nu$ sites.

10.3 Ajout de sites

Nous étudions dans cette partie l'ajout d'un ensemble de sites ne comportant pas de biais statistique. Le nombre de sites ajoutés est proportionnel au nombre de sites du document de départ. On note $\sigma \in [0,\infty[$ cette proportion.

Le nombre de sites de la partie x du document devient $n_x + n_x\sigma$ et le nombre de sites satisfaisant Φ_x devient $n_x\beta_x + n_x\sigma\mu_x$.

$$\begin{aligned} Test(n_x + n_x\sigma, n_x\beta_x + n_x\sigma\mu_x, \mu_x) &= 2e^{-2\frac{(n_x\beta_x + n_x\sigma\mu_x - n_x(1+\sigma)\mu_x)^2}{(1+\sigma)n_x}} \\ &= 2e^{-2\frac{n_x}{1+\sigma}(\beta_x - \mu_x)^2} \end{aligned}$$

On en déduit que l'intensité de la marque sera la même que si l'on avait tatoué un document contenant $\frac{n}{1+\sigma}$ sites.

10.4 Modification des sites

Nous nous intéressons à la modification d'un sous-ensemble des sites d'un document tatoué. Pour cette transformation, on choisit aléatoirement les sites qui seront modifiés. Ces sites sont modifiés de façon à vérifier Φ_0 et Φ_1 avec respectivement une probabilité de μ_0 et μ_1. On les modifie alors de sorte qu'une proportion $\tau \in [0, 1]$ d'entre eux vérifient Φ_x.

Le nombre sites de la partie x reste proche de n_x et le nombre de sites qui satisfont Φ_x dans cette partie est $n_x\mu_x\tau + n_x\beta_x(1-\tau)$.

$$\begin{aligned} Test(n_x, n_x(\mu_x\tau + \beta_x(1-\tau)), \mu_x) &= 2e^{-2\frac{n_x(\mu_x\tau + \beta_x(1-\tau)) - \mu_x)^2}{n_x}} \\ &= 2e^{-2n_x(1-\tau)^2(\beta_x - \mu_x)^2} \end{aligned}$$

La modification de sites fait diminuer la présence de la marque quadratiquement par rapport à la proportion de sites modifiés. On peut noter que la modification de sites revient à faire une suppression puis un ajout de sites non biaisés. C'est donc le cas où le schéma est le plus sensible.

Conclusion

L'intérêt principal d'une telle étude est d'estimer, connaissant le biais au départ, jusqu'à quel point on peut appliquer une transformation au document sans que la détection ne soit affectée. Elle permet aussi de mieux comprendre l'influence d'une modification de l'ensemble de sites sur la détection de la marque. On constate par exemple que la diminution de la marque est identique lorsqu'on supprime la moitié du document ou que l'on double le nombre de sites (en ajoutant des sites non-biaisés). D'autre part, notre schéma est relativement peu sensible à l'ajout et la suppression de sites. En revanche, la marque est très affectée lorsque l'on modifie des sites.

Chapitre 11

Protocole de détection génériques

Sommaire

Dans ce chapitre, nous présenterons deux protocoles de détection pour notre schéma générique. Ces deux exemples illustrent bien le type de travaux que l'on peut effectuer en partant du schéma générique. Le premier protocole permet de faire détecter la marque par un détecteur générique qui connait uniquement de la clé. À partir d'un tel protocole, il devient possible de construire un web-service de détection de la marque. Le second protocole a un cadre applicatif différent. Il permet à un propriétaire potentiel de convaincre un juge qu'un document lui appartient sans avoir à transmettre de clé.

11.1 Protocole de détection avec un détecteur générique

Dans cette section, nous montrons que le schéma générique peut servir à construire un protocole de détection lui aussi générique vis-à-vis du type de document tatoué. Ce protocole de détection permet à un utilisateur de vérifier si un document est tatoué sans connaissance de la clé et sans communiquer l'intégralité du document à vérifier. De plus, il est utilisable avec n'importe quelle implémentation du schéma de tatouage construit sur la base de notre schéma générique. Nous commencerons par définir chacun des acteurs avant de détailler les étapes du protocole.

11.1.1 Cadre applicatif

Pour des raisons de sécurité, on peut imaginer que le propriétaire d'un document ne possède pas la clé de tatouage, soit parce qu'il l'a détruite, soit parce qu'il a laissé un tiers tatouer le document. Cependant, il veut quand même laisser à certains utilisateurs le droit de vérifier si des documents lui appartiennent. Quand le propriétaire légitime veut vérifier si un document lui appartient (i.e. s'il est tatoué avec sa clé), il peut suivre la même procédure qu'un de ces utilisateurs.

11.1.2 Acteurs intervenant dans le protocole

Deux acteurs interviennent dans le protocole : le tiers de confiance et l'utilisateur qui veut vérifier si un document est tatoué. Pour chacun d'eux, nous détaillons son objectif et les ressources dont il dispose.

Le tiers de confiance ne connaît que la clé de tatouage

L'objectif du tiers de confiance est de permettre à l'utilisateur de vérifier si un document est tatoué par une certaine personne.

Cet acteur connaît la clé de tatouage de la personne en question. Par conséquent, lorsqu'on lui donne le codage d'un site, il est capable de calculer la partie à laquelle appartient ce dernier.

L'utilisateur, qui veut vérifier si un document est tatoué par un propriétaire donné

L'utilisateur possède un document sans savoir si celui-ci est tatoué ou non. Il ne connaît pas la clé de tatouage. Il va faire une demande au tiers de confiance pour savoir si un document est tatoué ou non.

Cet acteur est capable d'effectuer les opérations d'extraction des sites du document et de codage d'un site. Il peut aussi vérifier si un site satisfait Φ_0 ou Φ_1.

11.1.3 Explication du protocole

L'utilisateur va commencer par construire un bloc de données à partir du document qu'il désire vérifier. Il l'enverra ensuite au tiers de confiance qui utilisera ces informations et la clé pour décider si le document est tatoué ou non. Le tiers de confiance transmettra sa réponse à l'utilisateur. On peut donc résumer le protocole en quatre étapes :

1. calcul du bloc de données par l'utilisateur ;
2. envoi du bloc vers le tiers de confiance ;
3. décision du tiers de confiance ;
4. envoi de la réponse vers l'utilisateur.

Les étapes 2 et 4 consistent simplement à transmettre des données. Nous nous intéressons plutôt aux étapes 1 et 3. Dans cette section, nous verrons le détail de ces deux étapes.

Calcul du bloc de données par l'utilisateur

L'utilisateur possède le document à vérifier, il peut donc en extraire les sites. Il peut aussi calculer le codage d'un site et vérifier si celui-ci satisfait Φ_0 ou Φ_1.

L'utilisateur va extraire un certain nombre de sites du document en utilisant la fonction publique X. Le nombre de sites extraits est déterminé à l'avance. Pour chaque site s, l'utilisateur va utiliser les fonctions publiques C, T_0 et T_1 pour calculer un triplet $(C(s), T_0(s), T_1(s))$. Il enverra ensuite l'ensemble des triplets calculés au tiers de confiance. L'algorithme de la figure 13 reprend les actions effectuées par l'utilisateur.

Notons que dans cet algorithme, l'utilisateur construit un bloc de taille m, qui représente le nombre de sites du document à vérifier. Pour limiter les échanges sur le réseau, on peut facilement imaginer que l'utilisateur n'envoie qu'une fraction des sites du document.

Algorithm 13: Algorithme de calcul du bloc de données par l'utilisateur

Input: $d \in \mathcal{S}$: le document à vérifier, qui contient m sites
Output: l : la liste de triplets à envoyer au tiers de confiance
begin
  $l \leftarrow \emptyset$;
  for $i \in \{1, \ldots, m\}$ **do**
    $s \leftarrow X(i, d)$;
    $c \leftarrow C(s)$;
    $t_0 \leftarrow T_0(s)$;
    $t_1 \leftarrow T_1(s)$;
    Ajouter (c, t_0, t_1) à l ;
  return l
end

Décision du tiers de confiance

Avec l'ensemble de triplets envoyé par l'utilisateur, le tiers de confiance utilisera la clé secrète pour partitionner les sites et vérifier la présence d'un biais statistique dans les deux premières parties. L'algorithme de détection de la figure 14 détaille comment le tiers de confiance va décider si le document est tatoué.

Lorsque le tiers de confiance répond «oui», c'est qu'il a trouvé un biais statistique dans l'ensemble des informations envoyées. Par contre, si une réponse négative est renvoyée, cela peut s'expliquer soit parce que le document n'a pas été tatoué, soit parce qu'il n'y avait pas suffisamment d'information envoyée. Le dernier cas peut se produire quand un nombre insuffisant de sites impliqués dans le tatouage a été envoyé ou quand le document a subit trop de transformations.

Algorithm 14: Algorithme de décision du tiers de confiance

Input: l : l'ensemble de triplets $(C(s), T_0(s), T_1(s))$
Data: $k \in \mathcal{K}$: la perte de précision autorisée
Data: p : le nombre de parties de la partition
Data: λ : le seuil de décision
Data: μ_0, μ_1 : la probabilité qu'un site satisfasse respectivement Φ_0 et Φ_1 dans un document non-tatoué.
Output: *True* si l'ensemble de triplets comporte un biais, *False* sinon.

```
begin
    foreach (c, t_0, t_1) ∈ l do
        j ← P_p(c, k) ;
        if j = 0 then
            n_0 ← n_0 + 1 ;
            if t_0 = 1 then m_0 ← m_0 + 1 ;
        else if j = 1 then
            n_1 ← n_1 + 1 ;
            if t_1 = 1 then m_1 ← m_1 + 1 ;
    return Test(n_0, m_0, μ_0) · Test(n_1, m_1, μ_1) < λ
end
```

11.1.4 Fuite d'information

Peu d'information sur le document et la clé transitent sur le réseau. Lorsque le codage des sites ne permet pas de retrouver d'information sur le document, un espion qui écoute les échanges n'aura pas d'information sur la clé et le document original. On peut facilement se trouver dans ce cas de figure en choisissant d'implémenter le codage du site par un hachage des informations du site.

Notons tout de même que de l'information a été divulguée à l'utilisateur. À chaque réponse «oui», l'utilisateur sait que certains des sites qu'il a choisi sont impliqués dans le marquage. En réitérant la détection avec des sites bien choisis, on peut imaginer que l'utilisateur puisse apprendre quels sites sont impliqués dans le tatouage. Néanmoins, un tel algorithme serait difficile à développer.

Conclusion

Nous avons décrit un protocole de détection du document par un tiers de confiance générique. Ici, seul l'utilisateur à besoin de connaître l'implémentation des fonctions telles que l'extraction et le codage des sites. En d'autres termes, l'implémentation du tiers de confiance ne dépend pas du tout de la donnée à tatouer. On peut donc imaginer de développer un service Internet de détection de tatouage implémenté une fois pour toute et qui fonctionnerait pour tout algorithme issu de notre schéma générique.

Notons que le tiers de confiance n'effectue que peu de calcul pour chaque vérification. En effet, le temps de calcul effectués par le tiers de confiance est directement proportionnel au nombre de triplets envoyé par l'utilisateur. Pour réduire le temps de calcul, le tiers de confiance peut même paralléliser la boucle principale de l'algorithme.

11.2 Protocole de preuve de propriété

Cette section propose d'aborder un protocole pour un autre cas d'utilisation. Il permet au propriétaire d'un document de prouver à un juge qu'un document lui appartient sans divulguer la clé de tatouage.

11.2.1 Cadre applicatif

Ce schéma a été développé pour le cas d'utilisation suivant : un juge a trouvé un document suspect, il souhaite savoir si le document appartient à une certaine personne. La personne en question veut convaincre le juge que le document lui appartient mais elle ne veut pas lui divulguer sa clé.

Notons que le propriétaire potentiel pourrait tricher s'il avait connaissance du document à vérifier. Lorsque le juge soupçonne un cas de tricherie, il peut alors demander au propriétaire potentiel de lui donner sa clé. La clé du propriétaire sera alors diffusée uniquement lorsque le juge estime que c'est nécessaire.

11.2.2 Les acteurs intervenant dans le protocole

Deux acteurs interviennent dans le protocole : l'utilisateur et le propriétaire potentiel. Nous allons donner le but et les moyens de chacun.

Le propriétaire potentiel du document

Le propriétaire potentiel veut convaincre le juge qu'il est bien l'ayant droit d'un document sans diffuser la clé de tatouage. Il possède une clé secrète, mais ne connaît pas le document que le juge veut vérifier.

Le juge

Le juge possède un document à vérifier, il veut déterminer si un propriétaire potentiel est bien l'ayant droit d'un document.

Le juge connait les implémentations des fonctions publiques du schéma (X, C, T_0 et T_1) pour le type du document à tatouer. Ces fonctions lui serviront à calculer les codages de certains sites choisis et à vérifier s'ils satisfont Φ_0 ou Φ_1.

11.2.3 Explication du protocole

Le principe du protocole repose sur un jeu de questions/réponses. Le juge va envoyer une suite de questions au propriétaire potentiel dont il connaît déjà les réponses. Il n'y a que deux moyens de connaître les réponses à ces questions : soit en connaissant la clé, soit en connaissant le document. Nous supposons que le propriétaire potentiel ne peut pas accéder au document tatoué pendant l'exécution du protocole.

Si le propriétaire potentiel donne un nombre suffisamment important de bonnes réponses le juge pourra alors être convaincu qu'il a affaire à l'ayant droit du document.

Le protocole s'exécute en suivant les étapes suivantes :

1. calcul des informations à conserver et à transmettre par le juge ;
2. envoi des informations à transmettre vers le propriétaire potentiel ;
3. calcul des réponses par le propriétaire potentiel ;
4. envoi des réponses au juge ;
5. décision du juge.

Seules les étapes 1, 3 et 5 sont à détailler, les autres étant simplement des échanges d'information.

Calcul des informations à conserver et à transmettre

Le juge possède le document à vérifier et peut exécuter les fonctions publiques sur les sites. Ainsi, il est capable d'extraire et de coder un site. Il peut aussi vérifier si un site satisfait Φ_0 ou Φ_1. Le juge va choisir un certain nombre de sites du document et calculer deux listes. La première contient le codage de chaque site et la seconde représente l'ensemble des sites qui satisfont Φ_0 et Φ_1. L'algorithme 15 montre comment calculer ces deux listes. La première liste est transmise au propriétaire potentiel tandis que la seconde est conservée pour être utilisée lors de la dernière étape.

Calcul des réponses par le propriétaire potentiel

C'est à cette étape que le propriétaire potentiel utilise la clef de tatouage qu'il possède. Il va construire une liste qui représente, pour chaque codage reçu, la partie à laquelle doit appartenir le site. Si le codage appartient à la première ou la seconde partie, le propriétaire met dans la liste l'indice de la partie à laquelle le site appartient. Si le codage appartient à une autre partie, un symbole indiquera que ce site doit être ignoré.

L'algorithme 17 montre comment le propriétaire potentiel construit sa réponse. Cette réponse se présente sous d'une liste dans laquelle la valeur 0 (resp. 1) indique que le propriétaire légitime suppose que le site satisfait Φ_0 (resp. Φ_1). Dans l'algorithme, le symbole $\star$ indique que le site n'est pas significatif.

Algorithm 15: Algorithme de calcul des information à transmettre et à conserver

Input: $d \in \mathcal{S}$: le document à vérifier, qui contient m sites

Output: (l_c, l_t) : respectivement la liste des codages (à transmettre) et la liste des sites satisfaisant Φ_0 et Φ_1 (à conserver)

```
begin
    l_c ← ∅ ;
    l_t ← ∅ ;
    for i ∈ {1, ..., m} do
        s ← X(i, d) ;
        c ← C(s) ;
        t_0 ← T_0(s) ;
        t_1 ← T_1(s) ;
        Ajouter c à l_c ;
        Ajouter (t_0, t_1) à l_t ;
    return (l_c, l_t)
end
```

Algorithm 16: Algorithme de calcul des réponses du propriétaire potentiel

Input: l_c : la liste des codages transmise par l'utilisateur

Data: $k \in \mathcal{K}$: la perte de précision autorisée

Data: p : le nombre de parties de la partition

Output: l_a : la liste des réponses

```
begin
    l_a ← ∅ ;
    for c ∈ l_c do
        j ← P_p(c, k) ;
        if j = 0 then
            Ajouter 0 à l_a ;
        else if j = 1 then
            Ajouter 1 à l_a ;
        else
            Ajouter ⋆ à l_a ;
    return l_a
end
```

Décision de l'utilisateur

La décision du juge est basée sur la comparaison de la liste qu'il a conservée et de celle retournée par le propriétaire potentiel. En parcourant les deux listes, l'utilisateur va compter les «bonnes» réponses. Une réponse est comptée «bonne» lorsque le propriétaire potentiel a répondu 1 (resp. 2) et que le site satisfait Φ_0 (resp. Φ_1) dans le document à vérifier. Si ce nombre est significativement plus élevé que ce qu'obtiendrait quelqu'un ne possédant pas la clé, alors, le juge pourra être convaincu qu'il s'agit bien du propriétaire légitime.

Nous supposons que l'implémentation du schéma ne permet pas de savoir si un site satisfait Φ_0 ou Φ_1 à partir d'un codage sans avoir la clé. Supposons aussi qu'on a $\mu_0 = \mu_1$, ce qui est le cas pour les deux schémas présentés. Dans ce cas, lorsqu'il ne connaît pas la clé, un propriétaire potentiel qui répond 0 ou 1 a une chance sur deux de se tromper. Si on suppose que chaque question est indépendante des autres, le juge peut utiliser la borne de Chernoff pour calculer la probabilité que l'utilisateur ait donné ce nombre de bonnes réponses sans connaître la clé. Le juge décide du seuil à partir duquel il est convaincu. La décision du juge est donné par l'algorithme 17.

Algorithm 17: Algorithme de calcul des réponses du propriétaire potentiel

Input: l_t : la liste des sites qui satisfont de Φ_0 et Φ_1
Input: l_a : la liste des réponses transmises par l'utilisateur, de même taille que l_t
Input: $\lambda \in [0, 1]$: le seuil pour décider si l'on est convaincu on non
Output: $True$: si l'utilisateur est convaincu d'avoir à faire au propriétaire du document,
$False$ sinon

```
begin
    n ← 0 ; m ← 0 ;
    for i ∈ {0, ..., |l_t| − 1} do
        t_0, t_1 ← l_t[i] ;
        j ← l_a[i] ;
        if j = 0 then
            n ← n + 1 ;
            if t_0 = 1 then m ← m + 1
        else if j = 1 then
            n ← n + 1 ;
            if t_1 = 1 then m ← m + 1
    return Test(n, m, 1/2) ≤ λ
end
```

Discussion

Nous avons proposé un protocole qui permet de convaincre un juge que l'on est bien le propriétaire d'un document sans lui transmettre la clé. Si l'on substitue le juge par un utilisateur

dans lequel on n'aurait pas confiance, nous déconseillons d'appliquer le protocole. En effet, le propriétaire potentiel en renvoyant la liste de réponses divulgue beaucoup d'informations sur le partitionnement des sites. Ces informations pourraient être utilisées pour laver le document. Notons toutefois que cette fuite d'information est bien moins critique que pour un transfert de la clé. En effet, si la clé est directement transférée au juge, on divulgue alors la totalité du secret. De plus, on peut ne traiter qu'un sous-ensemble des sites, il faut alors répéter le protocole plusieurs fois pour connaître tous les sites impliqués dans le marquage.

Il faut aussi garder à l'esprit que le propriétaire potentiel puisse tricher s'il possède lui aussi le document tatoué. Pour lever toute ambiguïté, il faut toujours qu'il donne la clé au juge. Notre protocole peut néanmoins servir à décider s'il nécessaire d'aller jusqu'à l'échange de clé ou si le juge doit se déplacer chez le propriétaire potentiel pour effectuer la détection ou avoir un dispositif pour s'assurer qu'il ne peut pas utiliser le document lors du protocole (même si il le possède par ailleurs).

Conclusion

Dans ce chapitre, nous avons proposé un schéma aveugle, robuste, 0-bit et générique pour tatouer les données contraintes. Le schéma travaille localement sur les sites du document. Il insére une marque dans un document tout en préservant la qualité de celui-ci. La marque s'obtient par un biais que l'on introduit dans une propriété locale aux sites.

L'approche générique permet de travailler sur un schéma de tatouage indépendamment du type de données considéré. Nous avons ainsi montré qu'il est possible d'évaluer la transformation de l'ensemble des sites sur le document et de concevoir des protocoles génériques.

Un autre intérêt de l'approche générique est de faciliter la conception de schémas pour différents types de documents et pour différentes qualités à préserver. Pour valider cet aspect, nous avons donné des exemples d'implémentation pour les documents géographiques et pour les bases de données.

Conclusion

Contributions

Cette thèse s'inscrit dans le domaine du tatouage et plus particulièrement dans le domaine du tatouage de données contraintes. Dans cette thèse, nous avons donné deux contributions majeures ; la première concerne le tatouage de documents géographiques et la seconde le tatouage de données contraintes en général.

Le tatouage de documents géographiques vectoriels

Nous avons commencé par concevoir un algorithme de tatouage de données géographiques, robuste, 0-bit, aveugle et rapide. L'aspect le plus important du schéma et qui le différencie des autres schémas présentés dans l'état de l'art est de donner des garanties sur la préservation de la topologie et la métrique du document lors du tatouage. Nous pensons que la préservation de la qualité du document lors du marquage est un aspect fondamental pour les données géographiques dont les schémas existant n'ont pas assez tenu compte.

En plus de la préservation de la qualité, ce schéma est très efficace à la fois en temps de calcul et pour tatouer de petits documents. Grâce à notre schéma, il est possible de marquer et détecter la marque en quelques minutes sur des documents de tailles réelles. Rappelons aussi qu'il permet de tatouer des document très petits, qui contiennent donc peu d'information.

Les expérimentations poussées ont bien montré la robustesse de notre schéma contre le découpage ou le retatouage. Nous avons même proposé une variante qui améliore encore la robustesse du schéma face au découpage. On peut ainsi retrouver la marque dans des extraits de documents plus petits qu'avec le schéma original.

Le schéma générique pour les données contraintes

Notre seconde contribution est la conception d'un schéma de tatouage robuste, 0-bits, aveugle et générique. Ce schéma s'abstrait du type du document à tatouer et de la qualité de document. Il propose de travailler localement sur les *sites* du document. Le schéma que nous avons conçu présente de multiples avantages. Tout d'abord il ne demande pas de notion d'ordre, nous n'avons donc pas de problèmes de synchronisation pour retrouver la marque. D'autre part, il est particulièrement robuste à la suppression des sites du document. Enfin, le schéma est un guide qui

sera utile pour quiconque veut concevoir un schéma de tatouage pour de nouveaux types de documents ou de nouvelles qualités à préserver.

Bilan

Après tout ce travail, nous sommes maintenant persuadés qu'il est nécessaire d'avoir une solide expertise du type de données à tatouer pour concevoir un nouveau schéma, sinon le schéma a toutes les chances d'être inadapté. L'état de l'art des données géographiques nous a montré que le choix de certaines qualités à préserver ou des transformations considérées était très discutables et pourtant peu discutées. La définition de la qualité à préserver pour le schéma est en soit un travail délicat. Cette étape doit être opérée en concertation avec un expert du type de document à tatouer, idéalement le propriétaire qui souhaite utiliser le schéma. En effet, pour un même type de documents, on pourrait imaginer plusieurs contraintes à respecter. Cela dépend en grande partie de l'utilisation future que l'on souhaite faire du document et des aspects du document qui lui confèrent sa valeur.

Par ailleurs, nous sommes aujourd'hui arrivé à la conclusion que tatouage de données contraintes forme bien un domaine. Il existe une multitude de schémas de tatouage pour les données multimédia (image, son, vidéo). Par contre, les travaux sur les données structurés, et plus particulièrement vers les données pour lesquelles on garantit de préserver des contraintes formelles, sont plus rares. De tels schémas présentent pourtant l'intérêt de préserver la qualité, et donc la valeur (qu'elle soit pécuniaire, scientifique ou autre), d'un document lors du tatouage. Nous avons vu que les contraintes peuvent s'exprimer de différentes façons suivant le type de données à tatouer. Pour les base de données relationnelles, il s'agissait de préserver le résultat d'une requête de somme. Pour les données géographiques, préserver la qualité du document signifiait que l'on voulait préserver certains aspect topologiques du document et borner le déplacement des sommets.

Notre schéma générique reprend à la fois l'idée que les schémas puissent avoir des points communs tout en permettant de s'adapter aux spécificités du type de documents à tatouer. Il s'agit donc d'une excellente solution pour concevoir de nouveaux schémas pour les données contraintes.

Perspectives

Tatouage de données géographiques vectorielles

Nous avons vu que notre schéma pour les données géographiques vectoriels présente de multiples avantages. Le schéma a seulement été implémenté dans un programme indépendant. Il a cependant été suffisamment détaillé dans cette thèse pour que l'on puisse en donner une implémentation logicielle sous forme de module pour un ou plusieurs SIG.

Tatouage de bases de données relationnelles

Un approfondissement de cette thèse serait d'étudier plus en détails la robustesse de notre schéma pour les bases de données relationnelles. Il faudra alors reprendre les protocoles expérimentaux de D. Gross-Amblard et J. Lafaye pour pouvoir comparer les deux méthodes non plus qualitativement, mais quantitativement.

Tatouage de nouveaux type de données

Nous travaillons actuellement sur des schémas de tatouage pour d'autres types de données.

Nous sommes en train d'évaluer un algorithme de tatouage pour des colorations de graphe. Pour notre schéma, nous souhaitons préserver le graphe ainsi que le nombre de couleur de la coloration. Un site est défini pour chaque couple de sommet du graphe et contient ces sommets et leurs voisins dans le graphe. Nous avons conçu un codage qui consiste à compter les arêtes dans le voisinage de chacun des deux sommets. La propriété Φ est satisfaite si les deux sommets du site sont de la même couleur. Ce schéma est très intéressant car il est robuste à tous les isomorphismes du graphe.

En collaboration avec A. Widlocher, nous étudions l'opportunité de construire un schéma de tatouage supervisé pour la langue naturelle. Le principe du schéma serait de proposer à l'auteur des paraphrases qu'il validera ou non. Il s'agit là d'une application très différente et très difficile.

Si l'on reprend les terme les termes de notre schéma générique, la phrase constitue alors un site et l'auteur lui même valide la préservation de qualité du site.

Vers d'autres schémas génériques

Nous avons choisis certaines caractéristiques pour notre schéma générique. On pourrait cependant fixer d'autres choix pour concevoir d'autres schémas eux-aussi génériques.

Par exemple, notre schéma générique est un schéma 0-bit, il ne permet donc pas d'insérer un message dans le document. Nous pensons qu'il est possible de concevoir un schéma générique n-bits pour les données contraintes. Un tel schéma serait idéal pour pouvoir marquer le document en fonction du client qui l'achète, ce que l'on appelle le filigranage. On pourrait ainsi retracer l'origine de la fuite d'information.

Bibliographie

[Agrawal *et al.*, 2003] Agrawal, R., Haas, P. J. et Kiernan, J. (2003). A system for watermarking relational databases. *In SIGMOD '03 : Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 674–674, New York, NY, USA. ACM.

[Agrawal et Kiernan, 2002] Agrawal, R. et Kiernan, J. (2002). Watermarking relational databases. *In 28th International Conference on Very Large Databases (VLDB)*, volume 2, Hong Kong, China. IEEE.

[Alon et Spencer, 2000] Alon, N. et Spencer, J. H. (2000). *The Probabilistic Method*. Wiley-Interscience, seconde édition.

[Arnold, 2000] Arnold, M. (2000). Audio watermarking : features, applications and algorithms. *In Multimedia and Expo, 2000. IEEE International Conference*, volume 2, pages 1013–1016. IEEE.

[Atallah *et al.*, 2003] Atallah, M. J., Raskin, V., Hempelmann, C., Karahan, M., Sion, R., Topkara, U. et Triezenberg, K. E. (2003). Natural language watermarking and tamperproofing. *In IH '02 : Revised Papers from the 5th International Workshop on Information Hiding*, pages 196–212, London, UK. Springer-Verlag.

[Bazin *et al.*, 2008] Bazin, C., Bars, J.-M. L. et Madelaine, J. (2008). A novel framework for watermarking : The data-abstracted approach. *In IWSEC*, pages 201–217.

[Bazin *et al.*, 2007] Bazin, C., Le Bars, J.-m. et Madelaine, J. (2007). A fast, blind and robust method for geographical data watermarking. *In ASIACCS'07*.

[Bender *et al.*, 1996] Bender, W., Gruhl, D., Morimoto, N. et Lu, A. (1996). Techniques for data hiding. *IBM Systems Journal*, 35(3-4):313–336.

[Boney *et al.*, 1996] Boney, L., Tewfik, A. H. et Hamdy, K. N. (1996). Digital watermarks for audio signals. *In 1996 IEEE Int. Conf. on Multimedia Computing and Systems*, pages 473–480, Hiroshima, Japan. IEEE.

[Cayre *et al.*, 2004] Cayre, F., Devillers, O., Schmitt, F. et Maitre, H. (2004). Watermarking 3d triangle meshes for authentication and integrity. Research report 5223, INRIA, Unite de recherche INRIA Rennes, IRISA, Campus universitaire de Beaulieu, 35042 Rennes cedex, France.

[CGAL, 2010] CGAL (2010). Cgal, Computational Geometry Algorithms Library. http ://www.cgal.org.

[Chernoff, 1952] Chernoff, H. (1952). A measure of the asymptotic efficiency for tests of a hypothetis based on the sum of observations. *Ann. Math. Stat.*, 23:493–509.

[Coatrieux *et al.*, 2008] Coatrieux, G., Quantin, C., Montagner, J., Fassa, M., Allaert, F.-A. et Roux, C. (2008). Watermarking medical images with anonymous patient identification to verify authenticity. *In* Andersen, S. K., Klein, G. O., Schulz, S. et Aarts, J., éditeurs : *MIE*, volume 136 de *Studies in Health Technology and Informatics*, pages 667–672. IOS Press.

[Constantion *et al.*, 2005] Constantion, C., Gross-Amblard, D. et Guerrouani, M. (2005). Watermill : an optimized fingerprinting system for highly constrained data. *In ACM multimedia and security workshop*.

[Cox *et al.*, 2001] Cox, I., Miller, M. et Bloom, J. (2001). *Digital Watermarking*. Morgan Kaufmann.

[Cox et Miller, 2002] Cox, I. J. et Miller, M. L. (2002). The first 50 years of electronic watermarking. *EURASIP J. Appl. Signal Process.*, 2002(2):126–132.

[Dakowicz et Gold, 2006] Dakowicz, M. et Gold, C. (2006). Structuring kinetic maps. *In Progress in Spatial Data Handling*, pages 477–493, University of Glamorgan.

[Doncel *et al.*, 2005] Doncel, V. R., Nikolaidis, N. et Pitas, I. (2005). Watermarking polygonal lines using an optimal detector on the fourier descriptors domain. *In 13th European Signal Processing Conference - Eusipco2005*, Antalya.

[Doncel *et al.*, 2007] Doncel, V. R., Nikolaidis, N. et Pitas, I. (2007). An optimal detector structure for the fourier descriptors domain watermarking of 2d vector graphics. *IEEE Trans. Vis. Comput. Graph.*, 13(5):851–863.

[Douglas et Peucker, 1973a] Douglas, D. et Peucker, T. (1973a). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122.

[Douglas et Peucker, 1973b] Douglas, D. et Peucker, T. (1973b). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *In The Canadian Cartographer*, volume 10(2), pages 112–122.

[Drelie Gelasca *et al.*, 2005] Drelie Gelasca, E., Ebrahimi, T., Corsini, M. et Barni, M. (2005). Objective Evaluation of the Perceptual Quality of 3D Watermarking. *In IEEE International Conference on Image Processing (ICIP)*. IEEE.

[Fontaine et Galand, 2008] Fontaine, C. et Galand, F. (2008). How can reed-solomon codes improve steganographic schemes. *EURASIP Journal on Information Security*, special issue "Secure Steganography in Multimedia Content".

[Furon, 2006] Furon, T. (2006). A constructive and unifying framework for zero-bit watermarking. *CoRR*, abs/cs/0606034.

[Georges et Borouchaki, 1997] Georges, P. L. et Borouchaki, H. (1997). *Triangulation de Delaunay et maillage*. Hermes.

[Gross-Amblard, 2003] Gross-Amblard, D. (2003). Query-preserving watermarking of relational databases and xml documents. *In PODS 2003*, San Diego, CA.

[Guibas *et al.*, 1992] Guibas, L. J., Mitchell, J. S. B. et Roos, T. (1992). Voronoi diagrams of moving points in the plane. *In* Schmidt, G. et Berghammer, R., éditeurs : *Graph-Theoretic Concepts in Computer Science*, volume 570, pages 113–125. Springer.

[Gupta et Pieprzyk, 2007] Gupta, G. et Pieprzyk, J. (2007). Software watermarking resilient to debugging attacks. *Journal of Multimedia*, 2(2):10–16.

[Hembrooke, 1961] Hembrooke, E. (1961). *Identification of sound and like signals*. United States Patent 3,004,104.

[Hong *et al.*, 2002] Hong, D. G., Park, S. H. et Shins, J. (2002). A public key audio watermarking using patchwork algorithm.

[Huber, 2002] Huber, B. (2002). Gis & steganography - part 3 : Vector steganography.

[Kang *et al.*, 2007] Kang, H., Yamaguchi, K., Kurkoski, B. et Yamaguchi, K. (2007). Psychoacoustically-adapted patchwork algorithm for watermarking. *Intelligent Information Hiding and Multimedia Signal Processing, International Conference on*, 2:267–270.

[Kang *et al.*, 2001] Kang, L. H., Kim, K. L. et Choi, J. U. (2001). A vector watermarking using the generalized square mask. *In International Conference on Information Technology : Coding and Computing*, pages 234–236.

[Katzenbeisser et Petitcolas, 2000] Katzenbeisser, S. et Petitcolas, F. (2000). *Information Hiding*. Artech House Publishers.

[Kerkhoffs, 1883] Kerkhoffs, A. (1883). La cryptographie militaire. *Journal des sciences militaires*.

[Kitamura *et al.*, 2001] Kitamura, I., Kanai, S. et Kishinami, T. (2001). Copyright protection of vector map using digital watermarkingmethod based on discrete fourier transform. *In IEEE International Geoscience and Remote Sensing Symposium*, volume 3, pages 9–13.

[Kutter et Petitcolas, 1999] Kutter, M. et Petitcolas, F. (1999). A fair benchmark for image watermarking systems. *In Electronic Imaging '99. Security and Watermarking of Multimedia Contents*, volume 3657.

[Lafaye, 2007] Lafaye, J. (2007). *Tatouage des bases de données avec préservation de contraintes*. Thèse en sciences du cnam - spécialité informatique, Centre d'études et de recherches en informatique du CNAM.

[Lafaye *et al.*, 2007a] Lafaye, J., Beguec, J., Gross-Amblard, D. et Ruas, A. (2007a). Blind watermarking of geographical databases by polygon expansion. Rapport technique, CNRS-CCSD.

[Lafaye *et al.*, 2007b] Lafaye, J., Beguec, J., Gross-Amblard, D. et Ruas, A. (2007b). Geographical database watermarking by polygon elngation (technicl report). Rapport technique, Cedric, CNAM.

[Lafaye *et al.*, 2007c] Lafaye, J., Beguec, J., Gross-Amblard, D. et Ruas, A. (2007c). Invisible graffiti on your buildings : Blind & squarring-proof watermarking of geographical databases. *In SSTD'07*.

[Lafaye *et al.*, 2007d] Lafaye, J., Beguec, J., Gross-Amblard, D. et Ruas, A. (2007d). Protection des données géographiques par tatouage. *In SAGEO2007*.

[Lafaye *et al.*, 2007e] Lafaye, J., Gross-Amblard, D., Guerrouani, M. et Constantin, C. (2007e). Watermill : an optimized fingerprinting system for databases under constraints. *In IEEE Transactions on Knowledge and Data Engineering*.

[Li et Xu, 2003] Li, Y. et Xu, L. (2003). A blind watermarking of vector graphics images. *In Fifth International Conference on Computational Intelligence and Multimedia Applications*, pages 27–30.

[Lopez, 2002] Lopez, C. (2002). Watermarking of digital geospatial datasets : A review of technical, legal and copyright issues. *International Journal of Geographical Information Science*, 16:589–607.

[Ng et Lau, 2005] Ng, W. et Lau, H.-L. (2005). Effective approaches for watermarking xml data. *Database Systems for Advanced Applications*, pages 68–80.

[Nikolaidis *et al.*, 2000] Nikolaidis, N., Pitas, I. et Solachidis, V. (2000). Fourier descriptors watermarking of vector graphic. *In International Conference on Image rocessing*, volume 3, pages 10–13.

[Niu *et al.*, 2006] Niu, X., Shao, C. et Wand, X. (2006). A survey of digital vector map watermarking. *ICIC International*.

[Ohbuchi *et al.*, 2002a] Ohbuchi, R., Mukaiyama, A. et Takahashi, S. (2002a). A frequency-domain approach to watermarking 3d shapes. *In EUROGRAPHICS 2002*.

[Ohbuchi *et al.*, 2001] Ohbuchi, R., Takahashi, S., Miyazawa, T. et Mukaiyama, A. (2001). Watermarking 3d polygonal meshes in the mesh spectral domain. *In Graphics Interface 2001*, pages 9–17, Ontario, Canada.

[Ohbuchi *et al.*, 2002b] Ohbuchi, R., Ueda, H. et Endoh, S. (2002b). Robust watermarking of vector digital maps. *In International Conference on Multimedia and Expon (ICME'02)*, volume 1, pages 577–580, Lausanne, Switzerland. IEEE.

[Ohbuchi *et al.*, 2003] Ohbuchi, R., Ueda, H. et Endoh, S. (2003). Watermarking 2d vector maps in the mesh-spectral domain. *In Shape Modeling International*.

[Open Geospatial Consortium, 2002] Open Geospatial Consortium (2002). http ://www.directionsmag.com/article.php ?article_id=192.

[OpenStreetMap, 2009] OpenStreetMap (2009). http ://www.openstreetmap.org.

[Painter et A., 2000] Painter, T. et A., S. (2000). Perceptual coding of digital audio. *IEEE*, pages 451–513.

[Park *et al.*, 2002] Park, K. T., Kim, K. I. et Han, S. S. (2002). Digital geographical map watermarking using polyline interpolation. *In Advances in Multimedia Information Processing — PCM 2002*, volume 2532/2002, pages 225–243.

[Petitcolas, 2000] Petitcolas, F. (2000). Watermarking schemes evaluation. *Signal Processing Magazine, IEEE*, 17:58–64.

[Petitcolas *et al.*, 1998] Petitcolas, F. A. P., Anderson, R. J. et Kuhn, M. G. (1998). Attacks on copyright marking systems. *In Proceedings of the Second International Workshop on Information Hiding*, pages 218–238, London, UK. Springer-Verlag.

[Qu et Potkonjak, 1998] Qu, G. et Potkonjak, M. (1998). Analysis of watermarking techniques for graph coloring problem. *In ICCAD '98 : Proceedings of the 1998 IEEE/ACM international conference on Computer-aided design*, pages 190–193, New York, NY, USA. ACM.

[Raynal *et al.*, 2001] Raynal, F., Petitcolas, F. et Fontaine, C. (2001). Évaluation automatique des méthodes de tatouage. *Traitement du signal*.

[Rivest, 1992] Rivest, R. (1992). The md5 message-digest algorithm. RFC 1321, Network Working Group.

[Schulz et Voigt, 2004] Schulz, G. et Voigt, M. (2004). A high capacity watermarking system for digital maps. *In 2004 workshop on Multimedia and security, MM'Sec '04*, pages 180–186, Magdeburg, Germany. ACM.

[Solachidis *et al.*, 2000] Solachidis, V., Nikolaidis, N. et Pitas, I. (2000). Watermarking polygonal lines using fourier descriptors. *In IEEE International conference on Acoustics, Speech and Signal Processing*, volume 4, pages 1955–1958, Istanbul.

[Sonnet *et al.*, 2003] Sonnet, H., Isenberg, T., Dittmann, J. et Strothotte, T. (2003). Illustration watermarks for vector graphics. *In 11th Pacific Conference on Computer Graphics and Applications (PG'03)*, pages 8–10.

[Todorov, 2005] Todorov, T. (2005). Improving the watermarking process with usage of block error-correcting codes. Rapport technique, Université de Limoges.

[Voigt et Busch, 2002] Voigt, M. et Busch, C. (2002). Watermarking 2d-vector data for geographical information systems.

[Voigt et Busch, 2003] Voigt, M. et Busch, C. (2003). Feature-based watermarking of 2d-vector data. *In Security and watermarking of multimedia contents*.

[Voigt *et al.*, 2004] Voigt, M., Yang, B. et Christoph, B. (2004). Reversible watermarking of 2d-vector data. *In 2004 Workshop on Multimedia and Security*, pages 160–165.

[Voigt *et al.*, 2005] Voigt, M., Yang, B. et Christoph, B. (2005). High-capacity reversible watermarking of 2d-vector data. *In SPIE-IS&t Electronic Imaging*, pages 409–417.

[Wang *et al.*, 2007] Wang, K., Lavoué, G., Denis, F. et Baskurt, A. (2007). Three-dimensional meshes watermarking : Review and attack-centric investigation. *In International*

Workshop on Information Hiding, Lecture Notes in Computer Science, pages 50–64. Springer-Verlag.

[Watermill, 2009] Watermill (2009). http ://watermill.sourceforge.net/wiki/index.php.

[Yeo et Kim, 2001] Yeo, I.-K. et Kim, H. J. (2001). Modified patchwork algorithm : a novel audio watermarking scheme. *In Information Technology : Coding and Computing, 2001*, pages 237–242.

[Yvinec, 2007] Yvinec, M. (2007). 2d triangulations. *In CGAL User and Reference Manual. 3.3*.