Protection des données géographiques par tatouage [1]

Julien Lafaye[*] — **Jean Béguec**[*,†] — **David Gross-Amblard**[*,×] — **Anne Ruas**[†]

[*] *Laboratoire CEDRIC, Spécialité Informatique – CC 432*
Conservatoire national des arts & métiers
292 rue Saint Martin, 75141 PARIS Cedex 3, France
julien.lafaye(@)cnam.fr

[×] *Laboratoire Le2i, Faculté des Sciences*
Université de Bourgogne, BP 47870, 21078 DIJON Cedex, France
david.gross-amblard(@)u-bourgogne.fr

[†] *Laboratoire COGIT, Institut Géographique National (IGN)*
2/4 Avenue Pasteur
94 165 SAINT MANDE Cedex, France
anne.ruas(@)ign.fr

RÉSUMÉ. *Les techniques de tatouages (*watermarking*) sont cruciales pour la protection de la propriété intellectuelle. Nous proposons une méthode de tatouage de la couche bâti des bases de données géographiques vectorielles. Les marques dissimulées résistent aux filtres géographiques courants comme l'équarrissage ainsi qu'aux tentatives délibérées de suppressions. L'impact sur la qualité du jeu de données, définie comme une composition de la précision spatiale et de la qualité angulaire, est évalué expérimentalement.*

ABSTRACT. *Due to the ease of digital copy, watermarking is crucial to protect the intellectual property of rights owners. We propose an effective watermarking method for vectorial geographical databases, with the focus on the buildings layer. Embedded watermarks survive common geographical filters, including the squaring transformation, as well as deliberate removal attempts. The impact on the quality of the datasets, defined as a composition of point accuracy and angular quality, is assessed through an extensive series of experiments.*

MOTS-CLÉS : *Bases de données géographiques, équarrissage, tatouage robuste.*

KEYWORDS: *Geographical databases, polygon squaring, robust watermarking.*

1. Ce travail est partiellement financé par l'ACI Sécurité & Informatique TADORNE (2004-2007).

1. Introduction

Les systèmes d'information géographique (SIG) existent depuis maintenant 40 ans mais, encore aujourd'hui, leur domaine d'applications ne cesse de s'étendre. Pour le grand public, ce phénomène est visible à travers la généralisation du GPS et la mise en place de portails géographiques publics tels que Google Earth et GéoPortail. La plupart de ces applications géographiques reposent sur une base de données vectorielles (points, polylignes et polygones).

Obtenir de telles données à la fois précises et exhaustives requiert un investissement conséquent. Les facilités de duplication et de redistribution apportées par Internet, les copies et usages illicites menacent les fournisseurs de données ayant réalisé ces investissements. Du point de vue légal, les fournisseurs cadrent l'usage de leurs données à l'aide d'outils juridiques et de dispositions contractuelles. Du point de vue des mesures techniques, le tatouage (*watermarking*) est un outil dont l'utilisation permet de prouver sa paternité sur un contenu et donc de dissuader les personnes malveillantes de commettre des fraudes. Il consiste en la dissimulation d'une marque de propriété invisible dans la base de données. Nous proposons ici une méthode de tatouage pour la couche bâti des bases de données géographiques.

Afin de dissimuler la marque, une altération des données est indispensable. La difficulté est de trouver un compromis acceptable entre l'altération des données et la robustesse du tatouage (sa résistance aux attaques) : plus l'altération autorisée est grande, meilleure sera la robustesse de la marque. De manière générale, on tente de maximiser la robustesse pour une distortion maximale acceptable déclarée au préalable. Définir précisément les notions de distortion et de qualité d'un jeu de données est donc un prérequis au tatouage.

Toutes les applications géographiques ne reposent pas forcément sur des données précises. Par exemple, la précision spatiale n'est pas indispensable à la conception de cartes touristiques, pour lesquelles de fortes transformations sont appliquées de manière à améliorer la lisibilité des données. D'autres applications manipulent des objets géographiques dont les limites spatiales sont parfois floues, comme les forêts, les falaises ou les hauts-fonds. Cependant, une grande majorité des applications requiert des données précises, par exemple pour une exploitation automatique (recherche de services par proximité, navigation GPS, analyse spatiale des risques, etc.) De tels jeux de données sont toujours géoréférencés par rapport à un système de référence (par exemple, le World Geodetic System – WGS84, système de référence du GPS). La précision peut même être une exigence légale, comme pour la localisation des récifs dans les données nautiques IHO/SHOM. Dans ce cas, le maintien de la précision par l'algorithme de tatouage est essentiel.

Le bâti représente une grande partie du contenu vectoriel des bases de données géographiques (80% du jeu de données utilisé dans nos expérimentations), ce qui renforce l'intérêt d'une technique de tatouage adaptée à ce type de contenu. Les bâtiments réels présentent des régularités géométriques (murs parallèles ou perpendiculaires) qui doivent être répercutées sur les polygones de la base de données. Ils sont d'ailleurs

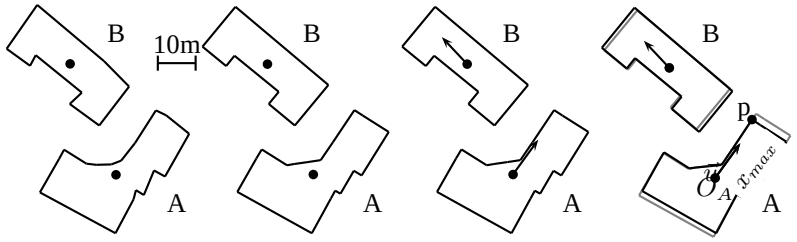
bien souvent corrigés après saisie de manière à refléter ces régularités. Cette correction, appelée *équarrissage*, améliore la qualité angulaire de la base de données tout en étant très invasive, car tout point de la base risque d'être déplacé. L'expérience montre qu'elle tend cependant à améliorer la précision spatiale moyenne. Ainsi, nous modélisons la qualité d'un jeu de données par sa précision *et* sa qualité angulaire (ce choix est renforcé par les conclusions d'autres travaux [NIU 06]).

Pour obtenir une méthode de tatouage robuste, il est indispensable d'anticiper les attaques potentielles. De ce point de vue, le tatouage de bases de données géographiques présente de réelles difficultés, puisque tous les utilisateurs, même légitimes, appliquent des transformations. Il existe plusieurs techniques récentes pour le tatouage des bases de données, numériques [AGR 03] ou géographiques [OHB 03, SCH 04, NIU 06]. Aucune de ces études ne prend en compte la transformation d'équarrissage, qui est pourtant appliquée quasi systématiquement par les utilisateurs.

Dans cet article, nous proposons une méthode de tatouage du bâti, robuste aux transformations classiques (équarrissage, simplification, lissage) ainsi qu'à une large classe d'attaques malveillantes. À notre connaissance, c'est la première méthode prenant en compte l'équarrissage. Elle offre un haut niveau de sécurité tout en gardant un impact borné sur la qualité (précision et qualité angulaire), et n'introduit pas d'erreurs topologiques (intersection de polygones). Notre méthode est aveugle (la base de données originale n'est pas nécessaire lors de la détection), et l'existence d'un indentifiant unique pour chaque polygone n'est pas requise. Elle est de plus incrémentale, ce qui permet de prendre en compte des bases de données volumineuses et leurs mises à jour.

Un schéma classique [AGR 03] des algorithmes de tatouage est d'introduire une corrélation secrète, pour chacun des objets du jeu de données, entre un identifiant robuste de cet objet (par exemple la clé primaire d'un n-uplet) et une de ses caractéristiques (par exemple la valeur d'un de ses attributs). La capacité à révéler cette corrélation sert de preuve de propriété. Dans notre approche, nous construisons un identifiant robuste pour chaque bâtiment en utilisant les coordonnées de son centroïde. Ensuite, nous nous appuyons sur le fait que les bâtiments sont généralement orientés, et que leurs arrêtes sont alignées ou perpendiculaires à cette orientation principale. Pour dissimuler un bit de marque, nous allongeons ou raccourcissons le bâtiment selon son orientation. Le coefficient d'élongation est choisi en fonction de l'identifiant du bâtiment, de la clé secrète du propriétaire, et du bit à dissimuler. Cette transformation est robuste, en particulier à l'équarrissage.

Structure de l'article Une introduction au tatouage, un modèle simple de bases de données du bâti et une définition formelle de la qualité sont présentés en section 2. Notre méthode de tatouage est décrite dans la section 3. La correction, la pertinence et la robustesse de la méthode sont mesurées en section 4. Les travaux connexes sont présentés en section 5 et la section 6 conclue.



(a) Originaux (b) Équarris (c) Orientation (d) Élongation

Figure 1. *Polygones représentant des bâtiments*

2. Préliminaires

2.1. *Qualité et filtrage des données géographiques*

Un point p est une paire de coordonnées (x, y) définis relativement à un système de référence R_0. Un polygone simple $P = (p_1, \ldots, p_n, p_{n+1} = p_1)$ est une liste de points (voir figure 1(a)). Une instance de base de données géographiques est une paire (R, DB) où R est le système de référence et $DB = \{P_i\}_{1,N}$ est un ensemble de N polygones. Un système de référence est associé à chaque instance afin de permettre son exploitation automatique. Notre méthode ne repose pas sur l'ordre des polygones dans la base, ni sur l'ordre des points dans un polygone, ni sur la présence de clés primaires identifiant les polygones.

La valeur (marchande) d'une base (R, DB) est liée à sa *précision moyenne*, sa *précision maximale* et sa *qualité angulaire*. La précision moyenne (resp. maximale) est la valeur moyenne (resp. maximale) de la distance entre un point de bâtiment (réel) et son correspondant dans la base. La qualité angulaire d'un polygone [AIR 96] est l'opposé de la somme des énergies des angles qui le composent. L'énergie d'un angle est une fonction quadratique et continue par parties, dont les minima sont atteints pour les multiples de 45°. Plus les angles d'un bâtiment sont proches de multiples de 45°, meilleure est sa qualité angulaire.

Le filtre d' *équarrissage* déplace les points des polygones de manière à rendre les angles presque droits parfaitement droits, contribuant ainsi à augmenter la qualité angulaire. La figure 1 montre deux polygones avant et après équarrissage.

2.2. *Tatouage*

Un schéma de tatouage est une paire d'algorithmes $(\mathcal{W}, \mathcal{D})$, où $\mathcal{W}$ est l'algorithme de tatouage et $\mathcal{D}$ l'algorithme de détection. L'algorithme $\mathcal{W}$ prend en entrée une base (R, DB), une clé secrète $\mathcal{K}$ et des paramètres de réglage, et retourne une base tatouée $(R, DB_{\mathcal{K}})$. L'objectif du détecteur est, étant donnée une base suspecte (R', DB') et

la clé secrète $\mathcal{K}$, de décider si la base est tatouée ou non. La procédure est dite aveugle si la base originale n'est pas utilisée par $\mathcal{D}$. Elle est robuste si l'altération raisonnable d'une base marquée n'affecte pas la détection de la marque. Un attaquant est libre d'utiliser une large palette d'attaques mais s'il souhaite tirer profit d'une copie illicite, il doit limiter la perte de qualité qu'il y introduit.

3. Tatouage du bâti

3.1. *Idée de l'algorithme*

Le modèle classique d'un algorithme de tatouage est d'introduire une corrélation secrète entre (1) une partie robuste des données, invariante aux altérations raisonnables, et (2) une caractéristique des données, dont l'altération est acceptable dans une certaine limite. C'est la mise en évidence de cette corrélation qui constitue la preuve de propriété. Dans cet article, un identifiant robuste id_i pour chaque polygone P_i est construit en utilisant les bits les plus significatifs des coordonnées de son centroïde, exprimées dans un système de référence R_0. L'intérêt d'un tel identifiant est d'être robuste à des altérations raisonnables des coordonnées des points du polygone. De plus, même si les coordonnées sont exprimées dans un autre système de référence R', différent de R_0, il est aisé de réaliser la conversion inverse pour appliquer la détection.

Afin de dissimuler un bit d'information dans un polygone, ce dernier est allongé ou raccourci selon son orientation. Cette orientation (voir figure 1(c)) représente l'angle majoritaire parmi les arêtes du polygone. Son calcul est détaillé dans la section 3.3. Par exemple, l'orientation d'une forme rectangulaire est parallèle à son plus grand côté. Le choix de cette orientation offre plusieurs avantages. Tout d'abord, la plupart des arêtes d'un polygone sont en relation de perpendicularité ou de parallélisme les unes par rapport aux autres. Lorsque le polygone est allongé suivant cette orientation, ces relations sont préservées. Ensuite, un tel allongement peut toujours être détecté même si le polygone a subi une rotation.

L'élongation doit être faite de façon à ce qu'un attaquant, connaissant l'algorithme de tatouage (mais pas la clé secrète utilisée), soit incapable deviner son coefficient. Une méthode classique est la suivante [AGR 03] : utiliser la concaténation de l'identifiant id_i du polygone avec la clé secrète $\mathcal{K}$ du propriétaire comme graine d'un générateur pseudo-aléatoire (GPA). Les tirages de ce générateur sont utilisés pour déterminer si le polygone courant doit être modifié et, si oui, avec quel coefficient d'élongation. La suite des tirages du générateur est imprévisible si la graine $id_i.\mathcal{K}$ n'est pas connue. Elle apparaît comme purement aléatoire à toute personne qui ne possède pas cette graine (un attaque peut facilement calculer id_i, mais $\mathcal{K}$ reste inconnue).

Exemple 1 *Un exemple de notre méthode de tatouage appliquée aux polygones A et B est présenté sur la figure 1(d). Les formes d'origine sont en noire, les tatouées en grisé. Tout d'abord, les centroïdes de A et B sont calculés, obtenant $O_A = (293, 155)$ et $O_B = (171, 447)$. Pour construire les identifiants id_A et id_B, les deux chiffres de*

poids le plus fort de chaque coordonnée sont concaténés, pour obtenir $id_A = 2915$ et $id_B = 1744$. Choisir ces deux chiffres convient à condition que le déplacement de points du plus de 10 mètres soit clairement déraisonnable, et que la distance typique entre deux bâtiments soit supérieure à 10 mètres (cette exemple est exprimé en base 10, alors que l'algorithme réel est en binaire). Ensuite, en se basant sur les sorties d'un générateur pseudo-aléatoire de graine id_A.$\mathcal{K}$, il est décidé que A doit être tatoué avec un bit de marque 0. L'orientation principale $\vec{u}$ est calculée ainsi que le point p tel que la longueur $x_{max} = \vec{u}.Op$ est maximale. Enfin, le polygone est allongé de façon à ce que x_{max} ait pour nouvelle valeur x^0_{max}, valeur prédéfinie encodant le bit 0. Le polygone B est traité de façon similaire. Le polygone A a ainsi été allongé, alors que B a été rétréci. Les angles des polygones demeurent inchangés par cette transformation.

Dans la suite, nous détaillons chacune des trois étapes successives de l'algorithme.

3.2. *Calcul des identifiants de polygones*

Nous utilisons les bits de poids fort des coordonnées du centroïde d'un polygone pour construire son identifiant. Notant h le plus petit bit significatif utilisé, il convient de choisir h suffisamment élevé pour que de petites modifications du polygone ne changent pas la valeur de l'identifiant. Parallèlement, h doit être choisi suffisamment petit de sorte que deux polygones adjacents ne partagent pas le même identifiant. Par manque de place, la méthode de calcul automatique de h est précisée dans un rapport technique [LAF 07a]. L'identifiant d'un polygone P est calculé en éliminant, dans les représentations binaires de ses coordonnées x et y, les bits qui représentent des puissance de deux au plus $h - 1$ et en concaténant les valeurs obtenues. Nous notons $hsb(O, h)$ cette opération : $id = hsb(O, h) = concat(hsb(x_O, h), hsb(y_O, h))$.

3.3. *Calcul des orientations des polygones*

Si $\vec{o}$ est un vecteur orientation, son poids dans le polygone P est défini comme la somme des longueurs des côtés de P qui sont parallèles à $\vec{o}$. L'orientation principale $\vec{u}$ du polygone est alors définie comme l'orientation ayant le plus grand poids. Pour supporter de petites imprécisions, nous considérons comme parallèles des arêtes qui le sont à une tolérance angulaire ε près. Dans un premier temps, nous créons un ensemble de k classes avec $2.\pi/k < \varepsilon$ et nous ajoutons, pour chaque i, les arêtes dont l'orientation est comprise entre $(i-1).\frac{\pi}{k}$ et $i.\frac{\pi}{k}$ à la classe i. Ensuite nous réduisons le nombre de classes en fusionnant les classes adjacentes dont le nombre de membres est différent de 0. L'orientation principale est alors obtenue comme la moyenne pondérée par les poids des arêtes contenues dans la classe ayant le plus grand poids. Un nombre de classes $k = 10$ donne de bons résultats en pratique. Cette méthode produit des résultats similaires à la proposition [DUC 03].

3.4. *Insertion d'un bit par élongation*

Nous présentons l'insertion d'un bit b dans un polygone P. Afin d'assurer la robustesse de l'insertion, nous altérons la forme générale du polygone. Plus précisément, nous modifions la plus grande distance x_{max} suivant l'orientation de P entre un sommet du polygone et son centroïde O. Si $\vec{u}$ est l'orientation principale, nous choisissons le vecteur $\vec{v}$ tel que $(0, \vec{u}, \vec{v})$ est une base orthonormale du plan. Le tatouage est alors effectué de la manière suivante :

– Calcul, pour chaque point p_i de P de l'abscisse x_i de p_i dans la base $(0, \vec{u}, \vec{v})$;

– Calcul de la longueur principale $x_{max} = \max_i |x_i|$;

– Modification des abscisses des points du polygone de telle sorte que que la nouvelle longueur principale devienne une des valeurs $\{x^0_{max}, x^1_{max}\}$ codant un bit 0 ou 1. Cette dernière opération est appelée *quantification*.

Étant donné un *pas de quantification* d, nous définissons les 0-quantificateurs (resp. les 1-quantificateurs) comme les valeurs $q^k_0 = k.d$ (resp. $q^k_1 = k.d + d/2$), pour $k \in \mathbb{Z}$. Afin de coder un 0, la valeur x_{max} sera transformée pour donner la valeur du 0-quantificateur le plus proche (idem pour coder un 1 sur les 1-quantificateurs). Le coefficient d'élongation du polygone est défini comme le rapport de la longueur quantifiée sur la longueur initiale $\sigma = {x'}_{max}/x_{max}$. Nous transformons alors tout point $p = x.\vec{u}+y.\vec{v}$ du polygone original en un point $p' = \sigma.x.\vec{u}+y.\vec{v}$ dans le polygone tatoué. La translation opérée sur chacun des points des polygones est au maximum $d/2$. Cette valeur n'est atteinte que pour les points les plus éloignés du centroïde du polygone suivant l'orientation principale. En moyenne, et pour ces points, l'amplitude de la translation est de $d/4$.

3.5. *Algorithme de tatouage*

L'algorithme 1 est paramétré par un entier γ représentant la période moyenne entre deux polygones tatoués. Pour chacun des polygones de la base, nous calculons son identifiant id que nous utilisons pour initialiser un générateur pseudo-aléatoire (GPA). Ensuite, nous utilisons les valeurs produites par ce générateur pour déterminer si le polygone doit être tatoué et, le cas échéant, quel bit b devra y être dissimulé.

Si le pas de quantification d est constant pour l'ensemble du jeu de données, les longueurs principales de tous les polygones tatoués deviennent des multiples d. Cette caractéristique est exploitable par un attaquant potentiel pour identifier les polygones tatoués et donc générer une attaque ciblée efficace. Afin de pallier cet inconvénient, nous utilisons un pas de quantification variable pseudo-aléatoirement choisi, pour chacun des polygones tatoués, dans un intervalle $[d_{min}, d_{max}]$. Les valeurs $[d_{min}, d_{max}]$ deviennent alors également des paramètres de l'algorithme. L'utilisation d'un GPA permet la dissémination des polygones tatoués de manière quasi uniforme dans le jeu de données. La localisation précise de ces derniers est impossible à deviner pour tout attaquant qui ne disposerait pas de la clé secrète. La robustesse du procédé est large-

ment dépendante du choix des paramètres γ, d_{min} et d_{max}. S'ils ne peuvent être fixés à priori pour l'ensemble des cas réels, les règles suivantes demeurent valides :

– Le compromis entre distortion ($\gamma \uparrow, d_{min} \downarrow, d_{max} \downarrow$) et robustesse ($\gamma \downarrow, d_{min} \uparrow, d_{max} \uparrow$) est inévitable. Les résultats des expériences réalisées et présentées dans la section 4 permettent néanmoins de guider le choix de ces paramètres.

– Si la précision (d'acquisition) du jeu de données original est β, alors la précision du jeu de données tatouées est $\beta + d_{max}/2$. La vente de ce dernier peut être encadrée par un contrat fixant cette nouvelle précision. Réciproquement, si un acheteur souhaite une précision $\beta' > \beta$, il est possible de lui fournir un tel jeu de données en choisissant $d_{max} < 2(\beta' - \beta)$.

– L'altération d_{max} admissible sur chaque polygone doit être plus grande que la précision d'acquisition. Dans le cas contraire, les distortions de tatouage pourraient être assimilées au bruit d'acquisition et supprimées par filtrage.

La technique d'insertion de bits par élongation ignore les relations topologiques (adjacence, recouvrement partiel ou total) entre bâtiments. Le choix d'ignorer ces contraintes topologiques lors du tatouage est volontaire ; nous préférons détecter et éventuellement annuler les modifications. En pratique, nous avons observé que choisir $d_{max} = 4m$ pour une base de précision $1m$ engendre très peu de collisions.

Algorithm 1: Algorithme de tatouage

Input: clé secrète $\mathcal{K}$, période de tatouage γ, h, intervalle de quantification $D = [d_{min}, d_{max}]$
Data: (R_0, DB) : base originale
Output: $(R_0, DB_{\mathcal{K}})$: base tatouée
foreach *polygone* P *dans* DB **do**
  $O \leftarrow$ `centroid`(P);
  $id \leftarrow$ `hsb`(O, h) ; `/* identifiant robuste` id `*/`
  `seed`$(G, \mathcal{K} \cdot id)$; `/* initialisation du GPA` G `avec` $\mathcal{K} \cdot id$ `*/`
  if `nextInteger`$(G) \mod \gamma = 0$ **then**
    `// Tatouage du polygone`
    $\vec{u} \leftarrow$ `orientation`(P) ; `/* orientation */`
    $x_{max} \leftarrow \max\{p \in P | \mathbf{Op} \cdot \vec{u}\}$; `/* longueur principale */`
    $d \leftarrow d_{min} +$ `nextFloat`$(G) . (d_{max} - d_{min})$; `/* pas de quantification */`
    $b \leftarrow$ `nextInteger`$(G) \mod 2$; `/* bit à insérer` b `*/`
    $x'_{max} \leftarrow$ `quantize`(x_{max}, d, b) ; `/* quantification */`
    $\sigma \leftarrow x'_{max}/x_{max}$; `/* facteur d'élongation */`
    `expand`$(P, O, \vec{u}, \sigma)$;
    if `testCollision()` **then**
      `rollback()` ;

3.6. *Détection*

La première étape de la phase de détection est de convertir la base suspecte (R', DB') dans le système de référence original R_0. L'algorithme de détection est en tout point similaire à l'algorithme de tatouage à la différence près qu'au lieu d'insérer des bits, les bits insérés lors du tatouage sont recherchés. Plus précisément, la détection se déroule comme suit. Les valeurs $d_{min}, d_{max}, h, \gamma$ et $\mathcal{K}$ utilisés pour la détection doivent être identiques à celles utilisées pour le tatouage. Pour chacun des polygones, le générateur aléatoire est initialisé avec la clé secrète $\mathcal{K}$ concaténée à l'identifiant du polygone. Si le polygone satisfait la condition de tatouage (i.e. $\texttt{nextInteger}(G) \mod \gamma = 0$), nous calculons le bit attendu b. Par ailleurs, nous calculons également le pas de quantification compris entre d_{min} et d_{max} et décodons le bit b' présent dans le polygone. Pour connaître le bit présent dans une valeur x quantifiée, il suffit d'examiner si cette valeur est l'un des 0-quantificateurs ou des 1-quantificateurs. Si x est une valeur réelle quelconque, nous calculons le 0-quantificateur (resp. le 1-quantificateur) x'_0 (resp. x'_1) le plus proche. Si $b = b'$, on dit qu'il y a correspondance. La conclusion sur la présence ou non d'une marque est prise en fonction de la valeur du taux de correspondance. Si ce taux est de 100% la marque est clairement détectée, s'il est de 50% la marque n'est pas détectée (cas d'une base non tatouée). Ainsi la marque est détectée lorsque le taux de correspondance ρ est suffisamment éloigné de $1/2$, c'est-à-dire lorsque $|\rho - 1/2| \geq \alpha$ où α est le seuil de détection fixé au préalable. Lorsque $\alpha = -\log(\delta/2)/2t$ avec t nombre de bits attendus, la probabilité de détections incorrectes (faux positifs) est au plus δ (preuve omise). Nous avons utilisé cette formule lors des expériences pour garder la probabilité de faux positifs inférieure à 10^{-4}.

4. Expériences

Les expériences ont été effectuées sur le bâti de Pamiers(09) extrait du produit BD TOPO® de l'IGN [Ins 02]. La partie sélectionnée comprend 4278 polygones ($35\ 565$ sommets), représentant aussi bien des zones urbanisées que des zones plus rurales. Les filtres/attaques suivants ont été testés :

SQ Équarrissage. La force de l'équarrissage est déterminée par la distortion maximale autorisée sur chaque sommet.

DP Algorithme de simplification de Douglas-Peucker [DOU 73]. Cet algorithme simplifie les contours des polygones en supprimant les sommets dont la contribution à la forme n'est pas suffisamment significative. Un seuil de distance de simplification d permet de contrôler la force du filtre.

AKH Algorithme de tatouage dans les bits de poids faibles basé sur [AGR 03]. Sa force est contrôlée par le nombre de bits de poids faible considérés.

GN Bruit gaussien de moyenne nulle et d'écart type d utilisé pour modifier chacun des points du jeu de données.

CA Attaque de fenêtrage. Seulement une portion rectangulaire des données est utilisées comme support de détection.

Tableau 1. *Robustesse aux attaques*

Filtre	Var. moy. précision.	Var. moy. énergie ang.	Robustesse
WM($_{dmin=3,dmax=4}$)	**0.013**	**0.02**	
CA	0	0	Oui
GN (d=0.2m)	0.18	6.19	Oui
GN (d=0.6m)	0.53	40.00	Oui
SQ (d=1m)	0.19	- 14.11	Oui
DP (d=2m)	N/A	- 0.18	Oui
GN (d=1m)	0.89	78.52	$\gamma < 50$
DP (d=5m)	N/A	110.63	$\gamma < 60$
ETR	N/A	-6.53	$\gamma < 30$
ETR (échelle = 25000)	N/A	-6.06	Non
ETR (échelle = 250000)	N/A	-0.03	Oui
CE (échelle=0.90)	1.06	0.16	Non
CE (échelle=0.95)	0.53	0.02	Oui
CE (échelle=1.0)	0	0	Oui
CE (échelle=1.05)	0.53	0.09	Oui
CE (échelle=1.10)	1.06	0.27	Non
AKH (lpow2=3)	0.14	26.73	
AKH (lpow2=2)	0.07	13.01	
AKH (lpow2=1)	0.04	4.93	
AKH (lpow2=0)	0.02	1.58	

ETR Chacun des polygones de la base est remplacé par un rectangle, contrôlé par une échelle cible supprimant les rectangles rendus illisibles.

CE Élongation d'un polygone en suivante son orientation.

WM Notre algorithme de tatouage.

Il est d'usage de considérer qu'une attaque est réussie si elle enlève avec grande probabilité la marque tout en introduisant une perte de qualité comparable à celle introduite par tatouage. Il est nécessaire d'étalonner les expériences en prenant comme référence, par exemple, la distortion induite par l'algorithme de tatouage.

Le tableau 1 résume nos expérimentations, et montre que seules des transformations extrêmement agressives, ne respectant pas la qualité des données, sont capables d'effacer la marque. Par exemple, un bruit gaussien avec 1 mètre de moyenne efface la marque mais altère la qualité des données bien plus que ne le fait l'insertion du tatouage. Notre tatouage résiste efficacement à l'équarrissage et à la simplification de Douglas-Peucker ainsi qu'au fenêtrage. On remarque également que l'algorithme inspiré de AKH introduit une perte de qualité supérieure à celle de notre méthode (le détail de ces expérimentations est disponible dans un rapport technique [LAF 07a]).

5. Travaux connexes

Un point récent sur les techniques de tatouage de bases de données vectorielles est présenté dans [NIU 06]. À notre connaissance, aucune des techniques disponibles ne prend en compte l'opération d'équarrissage, qui est pourtant systématiquement appliquée. Parmi ces travaux, [OHB 02] nécessite de plus la base de données originale pour la détection, alors que notre méthode est aveugle. [KAN 01, SAK 00] sont fragiles au fenêtrage et engendrent des agrégations de points visibles. [HUB 02, LOP 03, PAR 02] procèdent par ajout de points fictifs et lissage, ce qui a pour effet de modifier la forme des polygones du bâti. [VOI 02] modifie uniquement les bits de poids faible mais est fragile à l'ajout de points. Les techniques de transformation de domaines [GOU 05] appliquées aux lignes ne respectent pas les contraintes du bâti. Enfin, [SCH 04] est robuste à la transformation de Douglas-Peucker, mais nécessite de poséder des points clairement identifiés dans les données suspectes.

6. Conclusion

Cet article présente une méthode de tatouage aveugle des bases de données polygonales, particulièrement adaptée aux données du bâti par sa résistance à l'équarrissage. Les expérimentations proposées montrent que tout attaquant réussissant à détruire le tatouage doit en contrepartie diminuer grandement la qualité des données (leur précision et leur qualité angulaire). La méthode est implantée dans un logiciel libre et générique de tatouage de bases de données [LAF 07b]. Les extensions naturelles de ce travail sont le développement d'algorithmes de tatouage pour d'autres couches géographiques, avec en ligne de mire le tatouage simultané de ces différentes couches sans introduction d'incohérences entre niveaux.

Remerciements Nous remercions Eric Grosso pour son aide technique lors de nos expérimentations, ainsi que pour de nombreuses discussions.

7. Bibliographie

[AGR 03] AGRAWAL R., HAAS P. J., KIERNAN J., « Watermarking relational data : framework, algorithms and analysis. », *VLDB J.*, vol. 12, n° 2, 2003, p. 157-169.

[AIR 96] AIRAULT S., « De la base de données à la carte : une approche globale pour l'équarrissage de bâtiments (in French) », *Revue Internationale de Géomatique*, vol. 6, n° 2-3, 1996, p. 203–217, Hermes.

[DOU 73] DOUGLAS D., PEUCKER T., « Algorithms for the reduction of the number of points required for represent a digitized line or its caricature », *Canadian Cartographer*, vol. 10, n° 2, 1973, p. 112–122.

[DUC 03] DUCHÊNE C., BARD S., BARILLOT X., RUAS A., TREVISAN J., HOLZAPFEL F., « Quantitative and qualitative description of building orientation », *Fifth workshop on progress in automated map generalisation, ICA, commission on map generalisation*, avril 2003.

[GOU 05] GOU H., WU M., « Data hiding in curves with application to fingerprinting maps », *IEEE Transactions on Signal Processing*, vol. 53, n° 10, 2005, p. 3988–4005.

[HUB 02] HUBER W. A., « GIS and Stegranography - Part 3 : Vector Stegranography », avril 2002, http://www.directionsmag.com/article.php?article_id=195.

[Ins 02] INSTITUT GÉOGRAPHIQUE NATIONAL, « BD TOPO - Descriptif technique (in French) », december 2002, http://www.ign.fr/telechargement/MPro/produit/BD_TOPO/JT_Agglo/DT_BDTOPOPays_1_2.pdf.

[KAN 01] KANG H. I., KIM K. I., CHO J. U., « A vector watermarking using the generalized square mask », *Information Technology : Coding and Computing,*, avril 2001, p. 234–236.

[LAF 07a] LAFAYE J., BÉGUEC J., GROSS-AMBLARD D., RUAS A., « Blind Watermarking of Geographical Databases by Polygon Expansion », rapport n° hal-00137956, mars 2007, CNRS-CCSD HAL, Available online http://hal.archives-ouvertes.fr/hal-00137956.

[LAF 07b] LAFAYE J., GROSS-AMBLARD D., GUERROUANI M., CONSTANTIN C., « Watermill : an optimized fingerprinting system for databases under constraints », *submitted to TKDE, ,* 2007.

[LOP 03] LOPEZ VAZQUEZ C. M., « Method of inserting hidden data into digital archives comprising polygons and detection methods », November 2003, US Patent no. 20030208679.

[NIU 06] NIU X., SHAO C., WANG X., « A Survey of Digital Vector Map Watermarking », *International Journal of Innovative Computing, Information and Control*, vol. 2, n° 6, 2006, p. 1301–1316.

[OHB 02] OHBUCHI R., R.UEDA, ENDOH S., « Robust watermarking of vector digital maps », *Multimedia and Expo, 2002. ICME '02*, vol. 1, 2002, p. 577– 580.

[OHB 03] OHBUCHI R., UEDA H., ENDOH S., « Watermarking 2D Vector Maps in the Mesh-Spectral Domain. », *Shape Modeling International*, IEEE Computer Society, 2003, p. 216–228.

[PAR 02] PARK K. T., KIM K. I., KANG H. I., HAN S.-S., « Digital Geographical Map Watermarking Using Polyline Interpolation », *PCM '02 : Proceedings of the Third IEEE Pacific Rim Conference on Multimedia*, London, UK, 2002, Springer-Verlag, p. 58–65.

[SAK 00] SAKAMOTO M., MATSUURA Y., TAKASHIMA Y., « A scheme of digital watermarking for geographical map data », *Symposium on cryptography and information security*, Okinawa, Japan, janvier 2000.

[SCH 04] SCHULZ G., VOIGT M., « A high capacity watermarking system for digital maps », *MM&Sec '04 : Proceedings of the 2004 workshop on Multimedia and security*, New York, NY, USA, 2004, ACM Press, p. 180–186.

[VOI 02] VOIGT M., BUSCH C., « Watermarking 2D-Vector Data for Geographical Information Systems », *SPIE, Security and Watermarking of Multimedia Content*, vol. 4675, 2002, p. 621–628.