
Capteur cartographique

2.1. Motivations

Un capteur cartographique (i.e. une base de données cartographique couplée à un système de localisation – GPS) peut fournir un grand nombre d'informations statiques à un système d'aide à la conduite automobile relatives à la route mais aussi à l'environnement et à l'infrastructure proche ou « à venir ». Cette possibilité d'aller faire des requêtes en aval dans la cartographie numérique, pour annoncer au conducteur un virage en épingle à cheveux non visible sur la droite ou encore l'approche d'un carrefour dangereux est communément appelée « horizon électronique ». Il est ainsi possible par exemple, par l'utilisation d'un capteur cartographique, de connaître :

- La topologie de la route (i.e. le graphe routier et donc les prochains carrefours et le plus court chemin d'un point A à un point B).
- Des informations relatives à chaque route / portion de route : le type de route (urbaine, rurale, route nationale, autoroute,...), la vitesse limite, le rayon de courbure (de façon statique dans la base ou par calcul dynamique sur la topologie),...
- Des informations relatives à une position donnée (Point d'intérêt – POI) : l'emplacement des radars, des stations essences,...

L'utilisation d'aide à la conduite la plus couramment effectuée de nos jours d'un tel capteur embarqué au sein des véhicules est le guidage routier (Figure 103) : guider l'utilisateur sur le trajet à effectuer pour se rendre à sa destination¹ et l'aider par exemple, si besoin est, à trouver la station essence la plus proche de sa position. Les moyens techniques proposés de nos jours, permettent une aide « avancée » à la conduite en proposant un guidage (trajet) en temps réel : ce trajet est recalculé / corrigé si l'utilisateur n'emprunte pas le chemin proposé. Récemment, ces systèmes se sont vu améliorer par la prise en compte de l'état du trafic en temps réel et des statistiques d'utilisation des routes selon les heures de la journée. Ils proposent désormais une aide à la conduite encore plus avancée (intelligente ?) qu'elle n'était avant, en fournissant des trajets différents en cas de bouchon et selon l'heure de la journée.

Ces systèmes se veulent aussi des aides à la sécurité en permettant par exemple d'avertir le conducteur de ses éventuels excès de vitesse. Cette information issue de ces appareils autonomes, n'est malheureusement pas très fiable dû au fait que la vitesse est généralement calculée par rapport aux positions GPS successives, et que ces positions souffrent d'une certaine imprécision (de 10 mètres pour un récepteur GPS commun). La localisation du véhicule subit aussi cette imprécision fortement et il arrive souvent que le « GPS » mette un certain temps avant de « comprendre » que le véhicule n'a pas emprunté le bon itinéraire.

¹ Système qui est simplement appelé dans le commerce, et ce par abus de langage, GPS



Figure 103 - Ancienne et nouvelle interface de navigation du capteur cartographique TomTom : Les « GPS » se veulent de plus en plus réalistes

D'autres systèmes avancés d'aide à la conduite automobile ont été proposés ces dernières années dans la littérature. Ils utilisent des informations fournies par un tel capteur afin d'améliorer la sécurité de conduite. Par exemple, (Polychronopoulos, et al., 2005) utilise ce capteur pour l'aide au maintien du véhicule dans la voie via la prédiction de la géométrie à venir de la route. Dans le même ordre d'idée, les systèmes d'avertissement pour virage, se basant sur un capteur cartographique, proposent d'anticiper l'approche des virages afin de définir la vitesse appropriée, d'adapter l'éclairage des phares (Lauffenburger, et al., 2007), etc. Un autre facteur accidentogène se situe au niveau des carrefours. Ainsi (Revue, et al., 2002) proposent d'avertir le conducteur des risques potentiels à l'approche de carrefour en se basant toujours sur l'analyse du graphe routier courant.

Un panel des applications possibles de l'utilisation d'un capteur cartographique (seul) est proposé et résumé par la société Navteq (éditeur de base de données routière cartographique) dans le schéma de la Figure 104 ci-dessous.



Figure 104 - Applications possibles aux systèmes ADAS d'un capteur cartographique (source NAVTEQ)

Ainsi, il est clair qu'il est possible et nécessaire d'exploiter les informations d'un capteur cartographique afin de prédire l'environnement « statique » futur pour réaliser des applications d'aide avancée à la conduite automobile (en anticipant la géométrie de la route par exemple, l'approche d'un virage ou d'un carrefour). En effet, l'anticipation basée sur la connaissance de la topologie et des autres informations fournies par un capteur cartographique est primordiale pour créer des systèmes ADAS intelligents.

Dans notre cas, et comme précisé au chapitre précédent, notre système de détermination de la limitation de vitesse visuelle souffre de certains problèmes et peut ne pas connaître la limitation en cours, ou ne pas pouvoir prendre une décision sur la prise en compte ou non d'un panneau, ne connaissant pas le positionnement du véhicule sur la route. Il est donc nécessaire de « fusionner » les informations issues d'un capteur cartographique pour réaliser un système intelligent de détection des limitations de vitesse.

2.2. Etat de l'art

2.2.1. Projets ADAS utilisant la cartographie

Comme le souligne (Bastiaensen, 2004), qui dresse un état de l'art des projets ADAS utilisant un capteur cartographique, une cartographie digitale peut être utilisée comme :

- Un capteur prédictif et une source d'information additionnelle (pour améliorer l'interprétation, pour de la prévention des tournants, de la reconnaissance des panneaux routiers, des applications de navigation,...)
- Une mémoire
- Un filtre (par exemple, bloquer les appels entrants dans des zones accidentogènes)
- Un moyen de créer de nouveaux services

Les capteurs cartographiques ont donc un potentiel très important pour des applications de sécurité routière. Ainsi, depuis quelques années, de nombreux projets sur l'étude et la réalisation de système avancée d'aide à la conduite automobile utilisant un tel capteur ont vu le jour, tels que les projets :

- NextMAP (2000 - 2001) [Navteq, TeleAtlas, BMW, Fiat, Daimler Chrysler, Jaguar, Renault, ERTICO] : Les objectifs de ce projet étaient d'étudier la faisabilité technique et économique de la création d'une base de données cartographique enrichie spécialement pour des projets ADAS et de développer et proposer des extensions au standard GDF. Leurs conclusions sont qu'il est techniquement possible d'utiliser de telles bases cartographiques pour des projets ADAS et que cela permet d'avoir de réelles améliorations pour ces systèmes. [sources (Bastiaensen, 2004)].

- ActMAP (2002 – 2004) [ERTICO, TeleAtlas, Siemens VDO, BMW, Daimler Chrysler, NAVIGON, Fiat, Navteq] : Comme son nom l'indique le projet ActMAP (Actual and dynamic MAP) a pour but d'examiner, de développer, de tester et de valider des méthodes pour renseigner dynamiquement des bases de données cartographiques utilisées par des applications ADAS embarquées dans des véhicules. Ils ont validés leur approche par différentes applications ADAS dont notamment l'échange dynamique des changements de limitations vitesse et de topologie de la route à cause de zones de travaux. Afin de parvenir à leurs buts, ils ont notamment dû développer un format d'échange de données standardisé permettant de traduire les formats propriétaires des deux fournisseurs de cartes et d'utiliser ces données de façon transparentes dans leurs applications. [sources : (ERTICO, et al., 2005)].
- ADASIS (2001 – aujourd'hui) [ERTICO, Navteq, TeleAtlas, BMW, DaimlerChrysler, Ford, Honda, Mitsubishi, Nissan, Opel, PSA, Renault, Siemens, Toyota, Volkswagen, Volvo,...] : Ce projet initié par NavTech et qui est maintenant un forum de ERTICO est parti du constat que les bases de données cartographiques ne sont pas accessibles par d'autres applications que le système de navigation et que les données y sont enregistrées dans le format propriétaire de ce système de navigation. Et que, de plus, les systèmes ADAS ont besoins d'accéder et d'utiliser ces données cartographiques afin d'améliorer leurs performances ainsi que pour développer de nouvelles fonctionnalités. De ce constat, de multiples objectifs ont été définis, dont notamment :
 - a. Définir, entre les différents professionnels membres, un modèle et une structure de données ouverts et standardisés pour représenter les données cartographiques au voisinage de la position du véhicule.
 - b. Définir une API ouverte et standardisée pour permettre aux applications ADAS d'accéder à l'horizon électronique du véhicule et aux données relatives à la position du véhicule.
- PreVENT / MAPS&ADAS (2004 – 2007) : Ce sous-projet (MAPS&ADAS) du projet PreVENT (55 partenaires, 10 pays représentés, 15 sous-projets) est inspiré par les besoins identifiés lors du forum ADASIS en ce qui concerne l'utilisation de la cartographie numérique comme capteur primaire et ou secondaire pour les systèmes ADAS. Leur conclusion est que l'accès aux données cartographiques autres que celles réservées à la navigation nécessite la mise au point d'une interface normalisée afin d'éviter la mise en place de solutions spécifiques dépendantes des OEMs ou des fournisseurs d'applications. Cela permettrait de réduire les coûts de mise en œuvre et une commercialisation plus rapide. De plus, la production de cartes ADAS nécessite un approvisionnement en attributs ADAS. De nouvelles méthodes ont ainsi été présentées, pour l'acquisition d'attributs ADAS permettant de proposer un système rentable de cartes numériques précises et mises à jour régulièrement qui soient conformes aux conditions d'application des systèmes ADAS. L'application présentée comme preuve de conception était un avertisseur d'excès de vitesse et de points névralgiques. [sources : (Fallou, et al., 2008) et (Daimler, 2008)].

- SafeMap (2003 – 2005) [TeleAtlas, Navteq, Bast, LCPC, Renault, DaimlerChrysler, PSA, egis mobilite] : Le but de ce projet est d'estimer les possibilités de réalisation socio-économique de cartes numériques comprenant des données pour la sécurité routière en proposant notamment des fonctions de sécurité basée sur la cartographie numérique et en analysant les exigences en matière d'exactitude des cartes numériques. Un démonstrateur a été développé proposant des aides afin d'avertir le conducteur de certains dangers sur la route, comme les virages ou les points noirs, ainsi que certaines fonctions propres aux poids lourds. Leurs essais démontrent que les systèmes d'information présentent un effet positif sur le contrôle de la vitesse et sont globalement bien jugés par les usagers. [sources : (Fallou, et al., 2008)].
- EDMap (2001 – 2003) [General Motors, Ford, Toyota, Navteq, Nissan] : Ce projet vise à fournir une preuve de concept pour des applications simple de sécurité routière à base de cartographie numérique et plus particulièrement des spécifications des bases de données cartographiques ainsi que l'évaluation du projet ambitieux de réalisation de cartes très détaillées pour appuyer ces applications. Les principales applications étudiées dans ce projet sont l'assistance pour la vitesse en virage (alerte et intervention), l'assistance pour les panneaux stop (alerte et intervention), l'avertissement de feux de signalisation (alerte) et l'assistance au suivi de voie (alerte). Pour chaque application, sont étudiés : les implications cartographiques, des démonstrations, l'étude de faisabilité de la base de données cartographiques et les technologies sans fil et d'échange de données. [sources : (Fallou, et al., 2008)]
- CVIS (2006 – 2010) [Navteq, TeleAtlas, Autoroutes du Sud de la France, BMW, CNRS/Heudiasyc UTC, DaimlerChrysler, Fiat, Inria, Itempora, LCPC, Renault, Siemens, TNO, Thales, Volvo,...]
- SAFESPOT (2006 – 2010) [Navteq, TeleAtlas, Centre National de la Recherche Scientifique, CG Côtes d'Armor, Compagnie Financière et Industrielle des Autoroutes, Continental, IBEO, LCPC, Siemens, Renault, DaimlerChrysler, TNO, Volvo,...]

Ces deux projets ont en commun le fait de vouloir développer, entre autre, la communication véhicule / infrastructure (V2I) afin d'améliorer la « vue » du conducteur sur son environnement. Cette communication leur permet d'informer le conducteur d'informations dynamiques tels que la présence de zone de travaux, d'accidents, d'éventuels ralentissements, de variation de la vitesse limite,... Ces nouveaux concepts sont appelés « carte locale dynamique » (local dynamic map). [sources : (ERTICO, 2009)].

Ces projets réalisés depuis les années 2000 dénotent certains problèmes récurrents quand à l'utilisation d'un capteur cartographique pour la réalisation d'applications ADAS. Il n'existe pas de standard de données cartographiques entre les 2 grands fournisseurs (Navteq et TeleAtlas), certains projets ayant comme tâche, afin d'arriver à leur but, de créer une interface de traduction entre des données dans un format normalisé et celles de l'un ou des deux providers. Les applications les plus recherchées sont celles traitant des aides sur la topologie du réseau routier (ex : prévention des

virages) et sur les limitations de vitesse. Enfin l'un des futurs enjeux est de pallier le problème des données statiques en créant des bases de données cartographique localement dynamique en utilisant entre autre la communication V2I (véhicule à infrastructure).

2.2.2. Fusion cartographie / vision pour aide à la limitation de vitesse

La fusion des informations d'un capteur cartographique avec d'autres informations dynamiques (issues de capteurs extéroceptifs) n'a été que très récemment (et donc très peu) étudiée. Par exemple, (Revue, et al., 2002) fusionnent les informations issues d'une caméra, d'un radar et d'un capteur cartographique afin d'informer des risques de collisions avec d'autres véhicules aux approches des carrefours et des virages. De la même manière (Gehrig, et al., 2003) fusionnent les informations cartographiques pour orienter la recherche par caméra de facteurs accidentogènes aux approches de carrefour.

En revanche, l'utilisation d'un capteur cartographique couplé à un système visuel (une caméra) pour la reconnaissance des limitations de vitesse n'est quasiment jamais envisagée dans la littérature. Cela est sans doute dû au fait que l'information sur la limitation de vitesse est considérée soit comme déjà existante dans les bases de données cartographiques, soit au contraire pas du tout. Cependant ces dernières remarques ne sont pas très perspicaces. En effet, l'information statique d'une telle base n'est jamais conforme à la réalité dont les limitations peuvent changer. Il faut donc mettre à jour la base cartographique régulièrement pour suivre l'évolution des restrictions sur les routes. De plus, elle ne comportera jamais les limitations temporaires (en cas de travaux par exemple) ni les limites variables (panneau LED) qui sont de plus en plus utilisées pour réguler le trafic sur les routes. Cela dit, au regard des informations secondaires que peut fournir un tel capteur, par exemple, si on considère que la limitation de vitesse dépend le plus souvent du contexte géographique (zone urbaine, extra-urbaine,...) et sur les cas particuliers de conduite (voie d'entrée ou de sortie d'autoroute, approche d'une école,...), il est alors possible d'utiliser cette information fournie par le capteur cartographique pour déduire la limitation de vitesse dépendant de la situation courante et coupler / fusionner cette déduction (croyance) avec notre croyance dans la détection issue d'un autre capteur (une caméra par exemple).

Dans les rares travaux de la littérature traitant ce sujet (Lauffenburger, et al., 2007) (Janssen, et al., 2004) (Bahlmann, et al., 2008) (Nienhüser, et al., 2009), la fusion des deux sources (cartographie et vision) est effectuée dans le cadre de la théorie des croyances (la théorie de Dempster-Shafer). Dans de tels systèmes, chaque information issue des 2 capteurs se voit attribuer un coefficient de confiance (une masse de croyance). Ces 2 informations peuvent ainsi être fusionnées plus ou moins intelligemment en prenant en compte d'autres types d'informations (critères) issus du capteur cartographique ou de la vision (type de route, heure de la journée,...) donnant plus ou moins d'importance (i.e. ayant une croyance plus ou moins grande) aux détections de l'un ou de l'autre capteur selon le contexte.

Par exemple dans (Lauffenburger, et al., 2008) la masse de croyance de chacune des détections (les *éléments focaux*) de la vision correspond à leur niveau de confiance directement donnée par l'algorithme de détection visuelle (qui est elle-même calculée à partir de la confiance calculée par le réseau de neurone utilisé pour la classification du panneau et du nombre de fois que le panneau a été vu par le suivi temporel). Il est aisé de comprendre que la détection par vision puisse retourner plusieurs résultats (i.e. avec des fausses détections). Cependant, afin de pouvoir effectuer la « fusion » il est nécessaire que le capteur cartographique donne lui aussi plusieurs résultats (*éléments focaux*). Or un capteur cartographique ne donne généralement qu'une limitation de vitesse pour un segment routier. Ainsi (Lauffenburger, et al., 2007) définit les éléments focaux (i.e. les potentielles limitations de vitesse) du capteur cartographique comme étant ceux pouvant être retournés si le système s'était trompé (par exemple avec une erreur de positionnement). Par exemple si la navigation stipule que la limitation de vitesse est de 110 km/h, les éléments focaux seront 110, 50 et 70. La masse de croyance attribuée à chacun de ces éléments focaux dépend de divers critères (C1, C2,...) prenant en compte l'erreur de positionnement du GPS, le type de route emprunté,... (Figure 105). La masse totale de chaque élément (i.e. la fusion) est calculée via la *loi de fusion conjonctive* énoncée dans la théorie de Dempster Shafer. L'élément focal (i.e. la limitation de vitesse) ayant la plus grosse masse (i.e. dont on a la plus grande croyance) est validé comme étant la limitation de vitesse à appliquer.

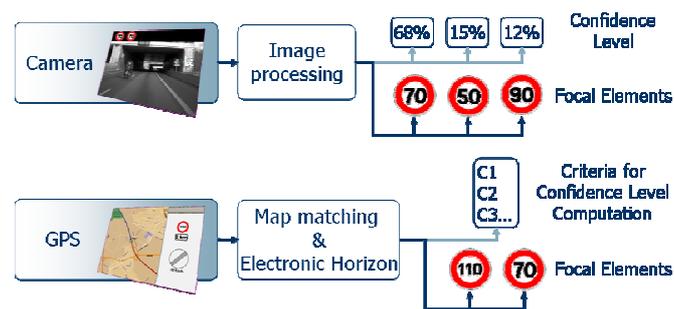


Figure 105 - Détermination des éléments focaux dans une approche de fusion par la théorie des croyances

Dans leurs travaux, (Nienhüser, et al., 2009) essayent de prendre en compte plus d'information dans le processus de fusion par la théorie des croyances en jouant sur le fait que les deux capteurs ont des comportements très complémentaires : la confiance à avoir dans chacun des capteurs doit dépendre des conditions d'éclairage. C'est ainsi qu'ils apportent plus de confiance aux éléments focaux issus de la vision quand il fait beau, quand une zone de travaux est détectée ou quand il s'agit d'un panneau à message potentiellement variable (panneau LED), tandis qu'ils accordent plus de confiance aux informations issues de la navigation dans les autres cas (de nuit, quand il y a du brouillard,...). Tous ces critères environnementaux sont évalués en temps réel à partir de la caméra pour agir sur le calcul des masses de croyance des divers éléments focaux.

Ces quatre travaux s'accordent tous à dire que l'utilisation unique d'un seul capteur (caméra ou cartographie) n'est pas suffisante : la cartographie ne prend pas en compte les zones temporaires (les travaux), les zones à messages variables (les panneaux LED) et les évolutions des limitations sur les routes, tandis que la caméra peut ne pas connaître la limitation de vitesse actuelle (quand un

panneau est non vu, en cas de fortes dégradations climatiques – brouillard, pluie forte,...). Ils concluent tous sur la nécessité de l'utilisation complémentaire des deux sources afin de fusionner leurs détections et d'en déduire la limitation de vitesse.

Il est donc clair qu'un capteur cartographique est d'une réelle nécessité dans notre système. La section suivante décrit les outils cartographiques existants pouvant être utilisés et intégrés dans un système ADAS.

2.3. Outils existants

2.3.1. Les Systèmes d'Information Géographique

Un système d'information géographique (SIG) permet d'organiser et de présenter des données spatialement référencées. Son rôle est de proposer une représentation plus ou moins réaliste de l'environnement spatial en se basant sur des primitives graphiques telles que des points, des vecteurs, des polygones ou des maillages,... À ces primitives sont associées des informations attributaires telles que la nature (route, voie ferrée, forêt, etc.) ou toute autre information contextuelle (nombre d'habitants, type ou superficie d'une commune par exemple). Typiquement, les SIG permettent d'effectuer des recherches sur les données attributaires basées sur la localisation (filtrage spatial) et vice versa, de rechercher seulement les informations géographiques qui ont certains attributs.

Il n'existe que 2 formats possibles pour la représentation des données dans les SIG :

- Le format matriciel (*raster model*) est la représentation de la zone géographique par un pavage où chaque cellule représente une valeur. Les cellules sont généralement représentées par des carrés (par exemple les pixels d'une image) mais en principe elles peuvent être de n'importe quelle forme. Par exemple, il est possible de représenter l'élévation d'une zone géographique par une image en niveaux de gris de cette zone où chaque valeur de pixel représente l'altitude (Figure 106). Pour un rendu plus réaliste, un SIG peut afficher par superposition plusieurs modèles matriciels (par exemple une orthophoto et un modèle numérique de terrain - Figure 107).
- Le format vectoriel (*vector model*) quant à lui contient des objets individuels qui représentent des phénomènes réels. Les composants spatiaux de chaque objet sont composés de points, de lignes, d'aires (des polygones) et de volumes. Seules les coordonnées des points particuliers (les extrémités des segments, les sommets des polygones,...) sont enregistrées dans le SIG faisant de cette représentation un format plus compact que le modèle matriciel (i.e. aucune information géographique n'est enregistrée là où il n'y a pas d'objet). L'aspect spatial d'une donnée vectorielle peut être traité comme un graphe avec toutes les propriétés et opérations définies dans le domaine de la théorie des graphes. Les

objets ont généralement des attributs associés (à l'instar de la valeur d'une cellule pour le format matriciel) permettant à l'utilisateur d'effectuer différentes requêtes sur les données.

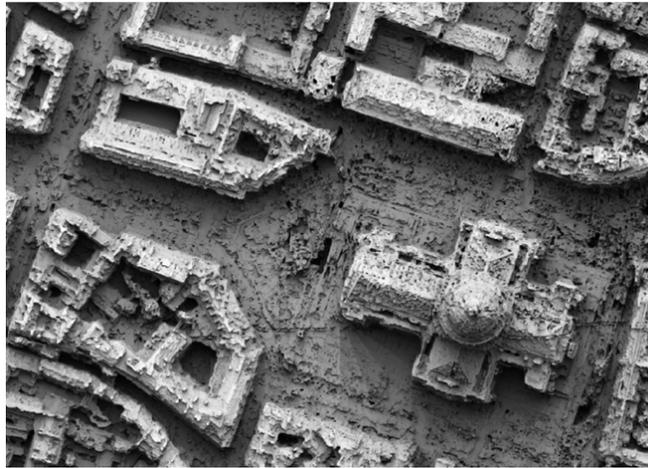


Figure 106 - Exemple de données matricielle : Modèle numérique de surface - MNS (source IGN projet TerraNumerica)

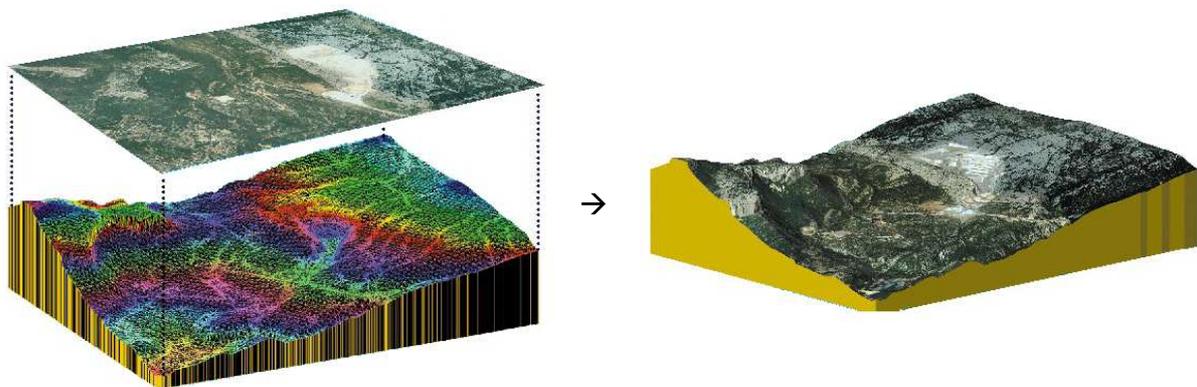


Figure 107 – Utilisation de plusieurs modèles matriciels : Modèle numérique de terrain (source projet TerraNumerica)

L'information géographique dans un SIG peut donc être représentée par cinq types de primitives différentes (Figure 108) : le point, la ligne, le polygone, le relief et les objets 3D.

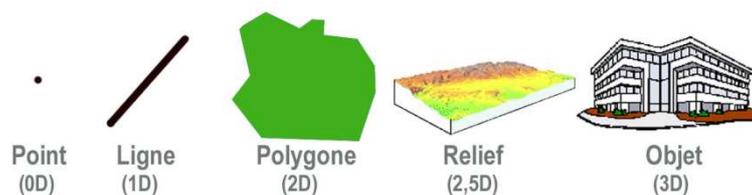


Figure 108 - Primitives d'un SIG

2.3.2. Les bases de données cartographiques routières

Ces bases de données représentent le réseau routier adjoint de caractéristiques. Elles sont couramment représentées (Figure 109) par un modèle vectoriel sous forme de graphe, comprenant des nœuds (points), des liens (des segments) et des aires (des polygones). Chacun de ces 3 éléments pouvant posséder des attributs (coordonnées, adresses, type de route, limitation de vitesse,...). D'autres éléments peuvent être inclus comme des points d'intérêts, la forme des bâtiments et les zones administratives (i.e. arrondissement, département, région, pays).

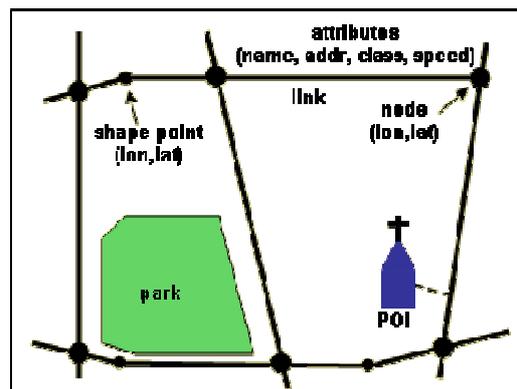


Figure 109 - Entités et attributs d'une base de données cartographique routière

Chaque nœud représente un point à la surface de la Terre et est représenté par une paire de coordonnées GPS latitude / longitude. Chaque lien représente un morceau de route entre deux nœuds. Un lien pouvant être un simple segment (correspondant à une portion droite d'une route) ou une courbe dont la forme est généralement décrite par des points intermédiaires (appelés *shape points*) le long de ce lien. Ces points intermédiaires sont représentés comme les nœuds par un couple de coordonnées GPS latitude / longitude mais ils n'ont pas pour but de représenter une interconnexion entre deux routes (à la différence des nœuds).

La question qui se pose est de savoir où trouver de telles bases en vue d'être utilisées et exploitées dans un premier temps au sein de ces travaux de recherche, puis pour être éventuellement intégrées facilement au sein des différents systèmes ADAS mis en œuvre au laboratoire.

Aux Etats-Unis, il est possible de récupérer gratuitement ce qui est appelé les *TIGER files* (*Topologically Integrated Geographic Encoding and Referencing*), des fichiers contenant le graphe routier des USA contenant certains attributs (nom, plage d'adresse,...) et couvrant tout le pays. Cette base de données a été entièrement développée par l'administration américaine, mais cette dernière ne fournit pas de logiciel pour les exploiter (il faudra donc se tourner vers des logiciels tiers ou redévelopper son propre logiciel afin d'exploiter ces données).

En Europe, il n'existe pour l'instant rien de similaire (i.e. des bases de données cartographiques mises gracieusement à disposition par les administrations) et il faudra donc se tourner vers des fournisseurs tiers spécialisés dans la création de base de données routière. Cependant cela risque de

changer dans un avenir proche. En effet, le parlement européen a publié le 25 avril 2007 dans le journal officiel la directive INSPIRE (Infrastructure for Spatial Information in the European Community). Celle-ci vise à établir à l'horizon de 2014, à harmoniser et à mettre à disposition par les états membres leurs informations géographiques et notamment des « réseaux de transport » (dont les réseaux routiers). Cela dit, cette directive n'impose pas de nouvelles collectes de données géographiques.

Les fournisseurs de cartes ne sont pas nombreux. Seules les sociétés Navteq et TeleAtlas fournissent des cartes routières détaillées au niveau international. Ces 2 sociétés ont été créées respectivement en 1985 et 1984. Elles ont toutes les deux été rachetées récemment : Navteq par Nokia en 2007 et TeleAtlas par TomTom également en 2007. L'historique de ces deux sociétés montre donc clairement que l'engouement dans la création de ces bases est un enjeu majeur pour la société d'aujourd'hui. Au niveau national (français), il n'y a guère que l'IGN (Institut Géographique National) qui crée et revend des bases de données géographiques.

Le principal problème de ces données acquises par ces sociétés réside dans le fait qu'elles coûtent très cher. Par exemple, l'IGN fournit la base de données routière GEOROUTE qui couvre la France entière pour la somme de 40 000 euros (Brunel, 2005). Navteq propose l'exploitation de ses bases de données pour 25 000 euros par an. On comprend aisément ces prix quand on sait que TeleAtlas y a investi en 20 ans près de 600 millions d'euros (Dessalle, 2006).

L'exploitation des données peut se faire de différentes façons. Il est possible de les acquérir directement auprès de ces 3 gros fournisseurs (Navteq, TeleAtlas ou l'IGN ; aux prix indiqués ci-dessus) qui transféreront leurs données (plusieurs giga octets) dans un format texte ouvert et lisible (généralement le format GDF - Geographic Data Files, norme ISO). Le projet européen INSPIRE semble aussi se baser sur ce format pour l'harmonisation des données géographiques des réseaux routiers. Malheureusement, ce format n'est pas fait pour être utilisé directement et il devra donc être nécessaire de retraiter ces données en vue d'être intégrées dans un système ADAS. Il est en revanche possible d'acheter auprès de revendeurs des données directement post-traitées munies d'un SDK (Software Development Kit) afin d'être exploitées facilement. Parmi les SDK les plus connus, on peut citer (avec leur prix pour en licence monoposte) : MapInfo (3,25k€), MacMap (5k€), AdasRP de la société Navteq (50k€),... L'inconvénient de ces SDK est simple : ce ne sont pas de vrais SDK indépendants, ils sont fournis sous forme d'une plateforme graphique de visualisation des données (Figure 110) qu'il est impossible de ne pas utiliser et sont donc des logiciels indépendants dans lesquels il serait possible d'ajouter des « plugins » développés soi-même, et non de vrais SDK intégrables dans des logiciels externes. Il en résulte qu'une intégration (propre) dans sa propre plateforme de développement n'est donc pas possible et rend la possibilité d'accès aux données très dépendant du SDK d'accès fourni avec ces données (par exemple il pourrait être difficile de retrouver certaines données selon certains critères, comme retrouver toutes les routes qui sont à moins de 100 mètres d'un point donné).

Le constat est simple : les données brutes coûtent très cher et les SDK des revendeurs ne sont pas appropriés au vu des limitations d'accès aux données induites par leur utilisation et au vu de l'impossibilité de l'intégration propre dans un logiciel tiers.

Ce constat avait aussi été réalisé en 2004 par Steve Coast qui lança le projet OpenStreetMap : un wiki de la cartographie où tout le monde peut ajouter des informations géolocalisées (des routes, des zones,...) dans une base de données communautaire. Depuis 2004, OpenStreetMap est en constante évolution et s'est enrichi de nombreuses données. L'intérêt de ce projet en Europe n'est plus à démontrer et c'est ainsi que, par exemple, la société néerlandaise Automotive Navigation Data, qui est un fournisseur local de cartographie aux Pays Bas, a gracieusement fourni sa cartographie des Pays-Bas. Certaines villes donnent aussi des fonds à OpenStreetMap afin qu'ils organisent des Mapping Party (weekend de rencontre entre utilisateurs, ou chacun se voit attribué une partie de la ville à géoréférencer) pour créer la cartographie de la ville.

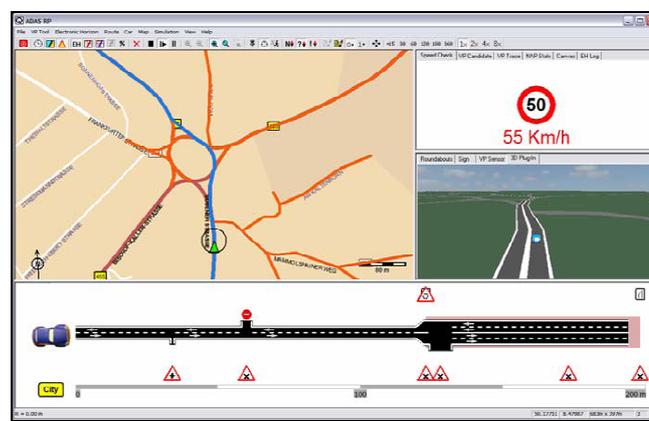


Figure 110 - SDK AdasRP et son interface graphique indissociable

La qualité des données issues d'OpenStreetMap par rapport à Navteq et TeleAtlas est relativement équivalente. En effet, ils ont la même façon de collecter les données (tous géolocalisent les routes du monde entier en les parcourant munis d'un récepteur GPS et en reportant ces informations dans leur base). Cependant Navteq / TeleAtlas prévoient un certain nombre d'attributs à collecter pour chaque route, OpenStreetMap au contraire n'impose rien aux utilisateurs, d'où une plus grande flexibilité et un plus grand nombre d'attributs possibles, mais l'information peut ou non être présente dans la base.

A ce jour, OpenStreetMap couvre les grandes villes d'Europe et d'Amérique du nord ainsi que le réseau routier constitué des autoroutes reliant ces villes.

Créer son propre capteur cartographique basé sur ces données en libre accès est donc tout à fait pertinent. En effet, cela permet de s'affranchir de beaucoup de problèmes : l'acquisition (le prix) des données brutes de Navteq ou TeleAtlas, ou l'utilisation non aisée des divers SDK du marché. L'avantage majeur d'un tel développement est de pouvoir définir son propre format de stockage des données géographiques avec sa propre API (SDK) d'utilisation et donc de pouvoir créer les fonctionnalités d'accès aux données à la demande. Enfin l'intégration dans divers logiciels, et notamment au sein de rMaps (notre logiciel de prototypage) en serait fortement plus aisée et plus

« propre ». C'est ainsi qu'est né le projet CAORTO au sein du laboratoire décrit dans la section suivante.

2.4. La CAORTO

2.4.1. Format

La CAORTO, la cartographie du CAOR (le Centre de Robotique de l'École des Mines de Paris), est ainsi définie comme étant une base de données cartographique routière se basant principalement sur des données gratuites et en libre accès (pouvant donc être mises à jour facilement). L'objectif est de concevoir une base de données du réseau routier en 2D (un graphe), et de fusionner dans le futur les travaux de l'équipe de numérisation 3D (MMS – Mobile Mapping System) du laboratoire pour y intégrer des informations 3D.

Bien sûr, le développement de ce nouveau capteur cartographique doit répondre aux problèmes des systèmes « concurrents » actuels, et vise donc à intégrer les données cartographiques du projet OpenStreetMap qui sont gratuites et en libre accès (sous licence *Creative Commons*), et fournir une API d'accès (le fameux SDK) simple d'utilisation et ouverte afin d'être facilement intégrable au sein de nos logiciels. Evidemment dans un souci de compatibilité et de portabilité l'ensemble de la CAORTO vise à être indépendante de la plateforme (donc pouvant fonctionner aussi bien sous Windows que sous Linux ou autre) et ne doit pas s'appuyer sur des bibliothèques externes (il n'est pas rare de trouver des outils développés mais qui ne fonctionnent plus car s'appuyant sur des bibliothèques externes n'existant plus). L'API d'accès aux données sera donc écrite dans un langage informatique de base (typiquement le C++) sans faire appel à des fonctionnalités avancées proposées par certains systèmes d'exploitation.

Les données OpenStreetMap ne sont pas diffusées sous le format GDF mais en XML (Extensible Markup Language), ce qui constitue des centaines de mégaoctets de données sous forme d'un fichier texte à traiter. La réalisation d'application ADAS, qui requiert évidemment que les algorithmes fonctionnent en temps réel (i.e. analyser les données dynamiques décrivant l'entourage de la voiture de façon très rapide) afin de fournir une aide quasi instantanée au conducteur, requiert donc un accès très rapide à l'information. Or, la recherche de données dans un fichier XML de grande taille est loin de répondre à cette contrainte. Il est donc nécessaire de post-traiter les données OpenStreetMap afin de définir son propre format de stockage du graphe routier.

Le schéma conceptuel de la CAORTO est décrit par la Figure 111. Un éditeur indépendant permet d'importer les données OpenStreetMap, enrichissant la base ou la mettant simplement à jour, et de modifier les données existantes de la base. Afin de rendre l'utilisation plus conviviale et plus aisée pour l'édition des données, cet éditeur repose sur une interface graphique. Là aussi, et pour répondre aux mêmes spécifications que l'API d'accès, cet éditeur est entièrement développé en Java qui propose nativement la création d'interfaces graphiques sans dépendance vis-à-vis du système

d'exploitation. L'API d'accès (sous forme d'une bibliothèque) permet d'interroger la base et de récupérer des données afin d'être utilisées dans nos logiciels. Elle joue l'intermédiaire entre la base et les logiciels et doit permettre une utilisation aisée et simple d'accès aux données.

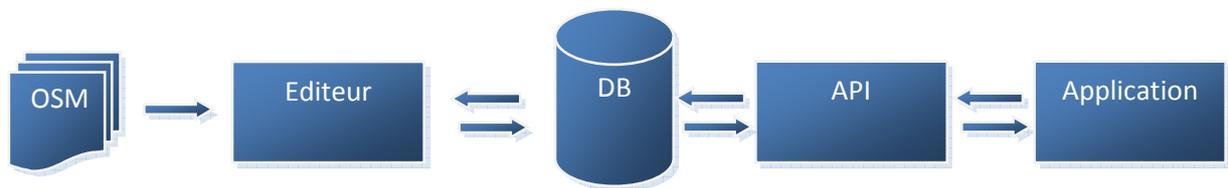


Figure 111 - Schéma conceptuel de la CAORTO

Toute la question réside dans le format de stockage des données dans la base. Il doit à la fois être facile à lire et rendre possible l'accès rapide aux informations quelque soit la zone du globe terrestre visée. La facilité de lecture par un opérateur humain n'est supportée que par les fichiers textes. Il ne faut donc pas escompter créer une base de données au format SQL où les données seraient enregistrées de façon binaire. La rapidité de recherche impose de découper et d'organiser intelligemment les données en plusieurs fichiers. Un format de stockage hiérarchique s'impose donc. Par exemple, Google utilise les QuadTree (une structure de données de type arbre dans laquelle chaque nœud peut compter jusqu'à quatre fils) pour accéder à n'importe quelle donnée de sa base géographique (des données au format matriciel, ici des images). Ainsi pour accéder à une zone sur le globe, il faut savoir à chaque niveau dans quel quart du planisphère la donnée se trouve et descendre successivement de niveau en niveau (Figure 112). L'intérêt d'une telle structure, en plus de la rapidité d'accès, est de permettre de stocker des informations macroscopiques au niveau des nœuds internes (par exemple les autoroutes) et microscopiques au niveau des branches (par exemple les petites ruelles).

La base géographique de la CAORTO n'est pas définie en QuadTree mais en « grille » (Figure 113) : chaque niveau de stockage est divisé en carrés de côté de 10° . Ainsi le premier niveau (le niveau mondial) qui fait 360° (+/- 180° de longitude) sur 180° (+/- 90° de latitude) comporte $36 \times 18 = 648$ pavés. Les niveaux inférieurs ne comportent eux que 100 pavés (10×10). La taille du plus bas niveau a été choisie empiriquement à 0.01° de côté, ce qui correspond à un petit quartier d'une ville. Les informations à y stocker ne seront donc pas nombreuses et il sera très rapide d'y accéder. Ainsi le monde est découpé sous forme d'un arbre à 4 niveaux.

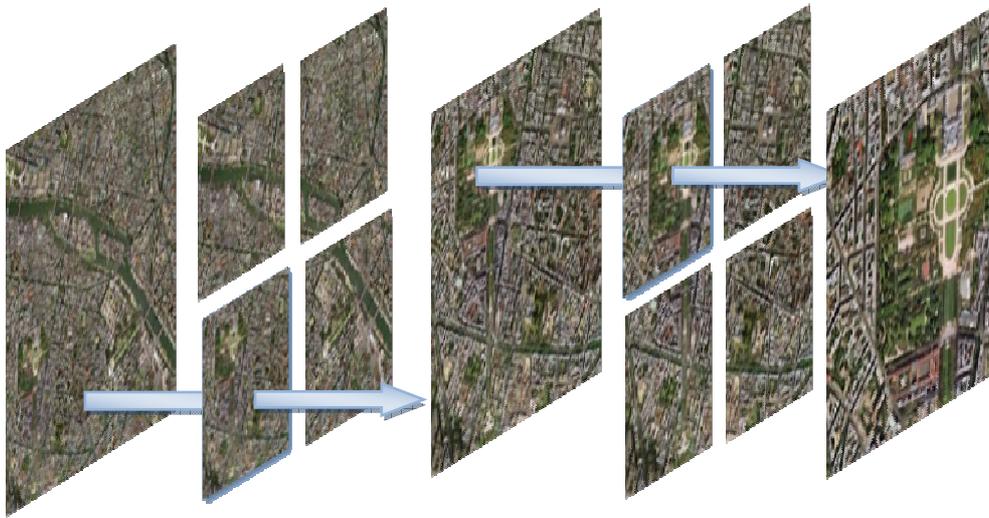


Figure 112 - Stockage sous forme de QuadTree des données matricielles chez Google Maps

D'un point de vue plus concret, chaque niveau est représenté physiquement sur le disque dur par un répertoire comportant les répertoires des niveaux inférieurs. Les derniers répertoires (ceux du niveau 4), contiennent le graphe routier (sous forme de fichier texte). Le nommage des répertoires et des sous-répertoires est corrélé avec le système de coordonnées latitude longitude (en degré) défini par l'arbre et se fait comme dans l'Exemple 1. Il est aisé de s'apercevoir que retrouver le bon répertoire où se situe une information se fait de façon quasiment instantanée (une dizaine d'opérations arithmétiques de base).

Au niveau d'une feuille est stocké le graphe routier de la zone en cours avec les attributs. Typiquement ces informations sont enregistrées dans 3 fichiers texte différents :

- un fichier référençant les nœuds, comportant pour chaque nœud un identifiant unique au sein de cette zone et son couple de coordonnées latitude / longitude ;
- un fichier pour les segments, comportant pour chaque segment un identifiant et le couple d'identifiants des 2 nœuds qui le définissent ;
- un fichier pour les attributs, comportant pour chaque attribut son type, sa valeur et l'identifiant du segment à qui il se réfère.

Le contenu de ces fichiers est complètement formaté, rendant leur lecture bien plus rapide (il n'y a pas de règle compliquée de parsing à appliquer comme il serait le cas avec le format XML générique). Par exemple, le contenu des fichiers de nœuds est ainsi sur chaque ligne : « id latitude longitude ».

Le poids total de l'ensemble de ces données au niveau d'une feuille n'est pas important. Par exemple le poids des données de la zone autour du jardin du Luxembourg, fait moins de 10ko répartis sur les 3 fichiers. Retrouver des informations dans 10Ko de données textuelles complètement formatées est aisément faisable en *temps réel* pour des applications d'aide à la conduite automobile.

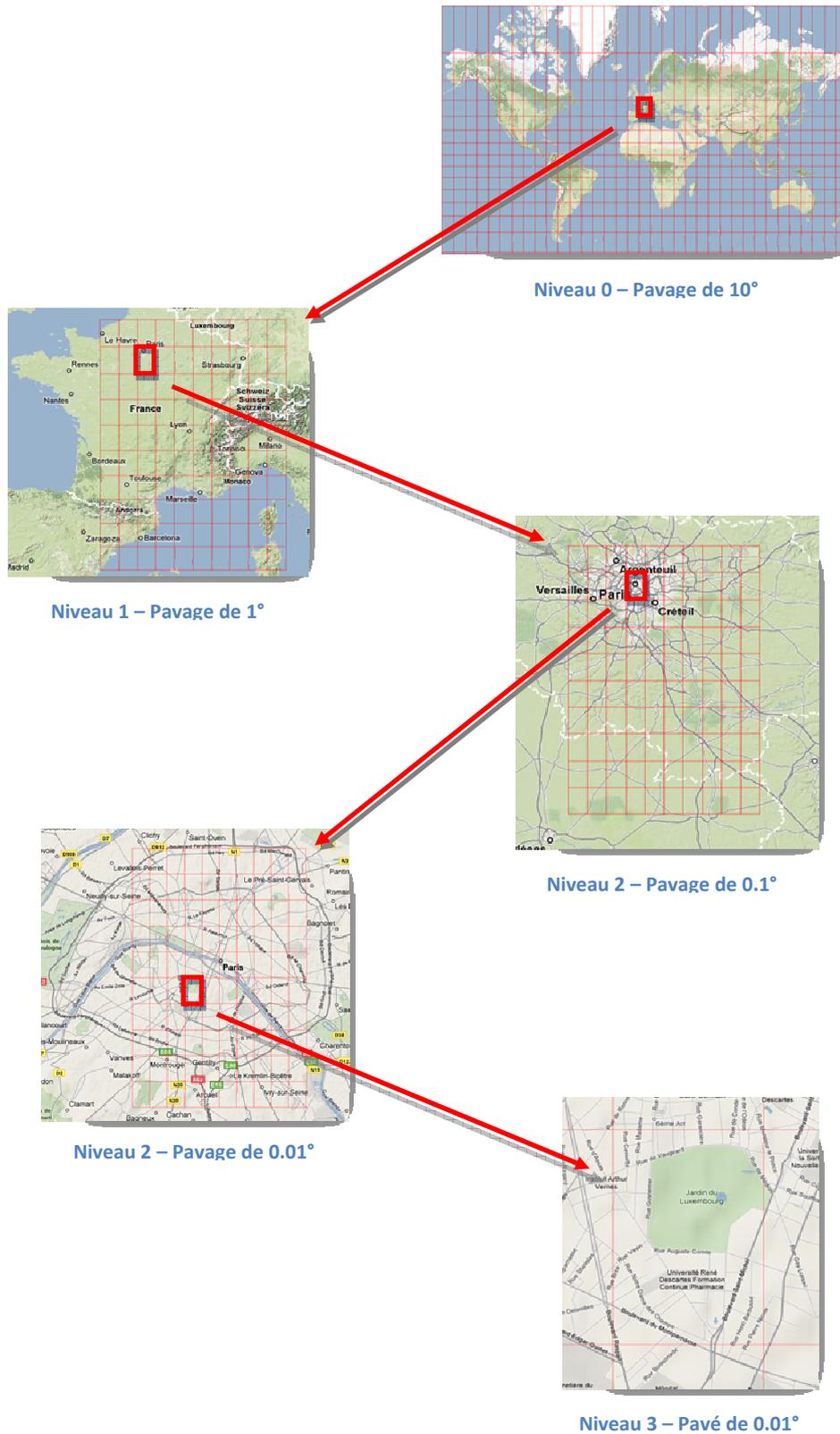


Figure 113 - Format hiérarchique de la CAORTO

On cherche une information d'un objet situé à la coordonnée (latitude, longitude) en degré de :

12.3456789 9.87654321

A noter que seuls les 2 premiers chiffres décimaux servent à l'indexation (puisque le plus grand niveau hiérarchique est constitué de carrés de 0.01 degré de côté), on peut donc retenir les coordonnées suivantes pour notre objet :

12.34 9.87

Il faut dans un premier temps normaliser ses valeurs entre de [-90;90] à [0;180] pour la latitude et de [-180;180] à [0;360] pour la longitude :

$$12.34 + 90 = 102.34$$

$$9.87 + 180 = 189.87$$

Le nom du 1^{er} répertoire est la juxtaposition "des dizaines" de longitude et de latitude :

$$102.34 / 10 = 10 \quad \rightarrow \quad \text{nom1} = 1810$$

$$189.87 / 10 = 18$$

Le nom du 2nd répertoire est la juxtaposition "des unités" de longitude et de latitude :

$$102.34 \% 10 = 2 \quad \rightarrow \quad \text{nom2} = 92$$

$$189.87 \% 10 = 9$$

Le nom du 3^{ème} répertoire est la juxtaposition "des dixièmes" de longitude et de latitude :

$$(102.34 * 10) \% 10 = 1023.4 \% 10 = 3 \quad \rightarrow \quad \text{nom3} = 83$$

$$(189.87 * 10) \% 10 = 1898.7 \% 10 = 8$$

Le nom du 4^{ème} répertoire est la juxtaposition "des centièmes" de longitude et de latitude :

$$(102.34 * 100) \% 10 = 10234 \% 10 = 4 \quad \rightarrow \quad \text{nom4} = 74$$

$$(189.87 * 100) \% 10 = 18987 \% 10 = 7$$

Finalement, l'arborescence de répertoire pour trouver notre objet est constituée des noms trouvés :

$$\text{db_path}\backslash\text{nom1}\backslash\text{nom2}\backslash\text{nom3}\backslash\text{nom4} \quad \rightarrow \quad \text{db_path}\backslash 1810\backslash 92\backslash 83\backslash 74$$

Exemple 1 - Accès une donnée dans la base CAORTO

Le problème du stockage du graphe routier sous forme d'arbre est qu'une rue peut chevaucher plusieurs zones (passer d'une zone à l'autre, comme c'est le cas du Boulevard Saint Michel sur la Figure 114). Il faut donc au sein d'une zone que les segments puissent faire référence à un nœud qui se trouverait dans une zone voisine. C'est ainsi que les identifiants des nœuds peuvent être de 2 types :

- interne, représenté par un nombre compris entre 0 et 100000000 (non inclus)
- externe, représenté par un nombre supérieur ou égal à 100000000

Le choix d'une limite à 100000000 étant une limite arbitrairement choisie. Un identifiant externe référence à la fois un objet d'un répertoire et le répertoire lui-même, tandis qu'un identifiant interne ne référence qu'un objet d'un répertoire (on suppose donc que les références se font pour des objets du même répertoire). Ainsi, lors de la lecture d'un fichier, savoir si un identifiant réfère à un objet d'une même zone ou non revient à effectuer un test sur sa valeur.

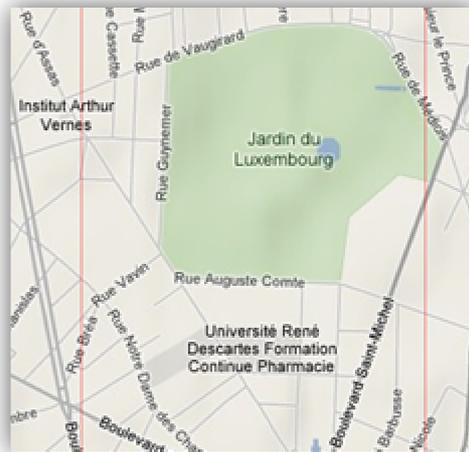


Figure 114 - Exemple de route passant par 2 zones différentes (limite des zones en rouge)

Dans l'Exemple 2 schématisant un fichier de segments de la base, le premier segment, d'indice 0, est délimité par les nœuds d'indice 0 et 1 de la même zone (du même répertoire). En revanche le second segment, d'indice 1, est délimité par le nœud 1 du même répertoire et par le nœud 8 d'un répertoire voisin identifié par « 1813282838 » (le début du nombre) correspondant au répertoire « db_path\1813\28\28\38\ ». La correspondance identifiant / chemin du répertoire est donc là encore très aisée.

0	0	1
1	1	181328283800000008

Exemple 2- Exemple de contenu d'un fichier de segment de la base CAORTO

2.4.2. Fonctionnalités

a) *Présentation générale*

Les fonctionnalités d'utilisation sont au cœur de l'API d'accès, car c'est celle-ci qui va dialoguer avec la base et récupérer les données selon les requêtes de l'utilisateur. Les principales fonctionnalités utiles pour un système ADAS sont :

- de pouvoir récupérer le segment (la route) sur lequel on se situe (i.e. le map-matching);
- retrouver les attributs (vitesse, nombre de voies,...) d'un segment (par exemple le segment en cours) ;
- de pouvoir se « balader » virtuellement de nœud en nœud par rapport au segment courant (i.e. pouvoir récupérer la position de la prochaine intersection qui peut être un carrefour ou tout simplement l'embranchement avec une voie de sortie sur une autoroute) ;
- de pouvoir retrouver les segments (les routes) dans un périmètre donné autour d'une position
- de pouvoir calculer des distances entre 2 coordonnées GPS (la distance donnant l'indication de risque)

Le routing (trouver le plus court chemin entre 2 positions selon divers critères) n'est pas considéré ici comme fonctionnalité élémentaire du système, cette fonctionnalité étant depuis de nombreuses années proposée dans les systèmes commerciaux du marché. Mais il serait aisé d'en développer les fonctions dans l'API.

b) *Map-matching*

En ce qui concerne le map-matching, fonction qui est la base d'un capteur cartographique, la littérature foisonne de recherches en la matière et pourrait constituer une thèse entière. En effet, le positionnement GPS est très important pour les applications ADAS, or cette technique requiert un ciel dégagé (pour la visibilité des satellites) et ne fournit une position correcte qu'à 10 mètres. Cependant, dans les grandes métropoles, en raison par exemple de la présence des hauts immeubles, la visibilité des satellites est très compromise. (Meng, et al., 2002) rapportent par exemple une disponibilité du système GPS de seulement 30% à Hong Kong. C'est pourquoi, il n'est pas rare de coupler un capteur GPS avec un odomètre en amont d'un algorithme de map-matching (Abuhadrous, 2005) ou de gyromètres (Jabbour, et al., 2008) . Un état de l'art complet du map-matching est proposé dans la thèse (Quddus, 2006). Ces algorithmes vont d'un matching assez simple (position du nœud le plus proche ou du segment le plus proche – matching géométrique), en passant par un matching un peu plus intelligent (calculer l'orientation du véhicule à partir de plusieurs points successifs et matcher le segment le plus proche ayant la bonne orientation – matching topologique) jusqu'à utiliser des concepts plus avancés en se basant sur des modèles de filtrage des données comme le filtre de Kalman, la théorie de Dempster-Shaper, la logique floue,... Cependant, (White, et al., 2000) rapporte que la plupart des algorithmes fonctionnent correctement sur autoroute et que toute l'intention doit se porter sur les problèmes d'intersection.

L'utilisation d'un GPS non « conventionnel » au sein du laboratoire de type DGPS (notamment utilisé par l'équipe de numérisation 3D), permet de s'affranchir quelque peu des problèmes de positionnement GPS dont la précision théorique est de 1 mètre (en urbain on note plutôt une précision de 2 à 3 mètres). C'est pourquoi, le map-matching proposé ici dans la CAORTO se base sur un matching géométrique retournant le segment le plus proche. Cette opération est basée sur une fonction de calcul de distance par rapport à tous les segments situés dans la zone courante (dans l'arbre de stockage) de la position actuelle ainsi que ses 8 zones voisines. Basée sur la même fonction, l'API propose de retrouver tous les segments situés dans un certain périmètre d'une position donnée. Bien entendu, il est possible de récupérer la position GPS projetée (corrigée ?) sur le segment map-matched.

L'API calcule en interne, et ce de façon automatique, l'orientation du véhicule (par différentiation des positions GPS successives). Cela permet de proposer la fonctionnalité de récupérer la coordonnée de la prochaine intersection en parcourant le segment courant (map-matchée) dans la bonne direction jusqu'à atteindre un nouveau nœud (et non un shape-point, la différentiation se faisant sur le nombre de segments connectés à ce nœud, i.e. le degré du sommet dans la théorie des graphes).

c) Calcul de distances

La fonctionnalité la plus importante d'un capteur cartographique est de pouvoir calculer la distance entre 2 coordonnées GPS. Le problème réside dans le fait que la terre n'est pas plate mais ronde. Ainsi la distance orthodométrique (aussi appelé distance du grand cercle) est souvent utilisée pour calculer la distance entre 2 points à la surface de la terre (Équation 2). Cette équation est cependant sujette à des erreurs d'arrondis dans le cas de calculs de petites distances. A la place de celle-ci, on lui préfère donc la distance d'Haversine (Équation 3) qui est plus juste en cas de petites distances.

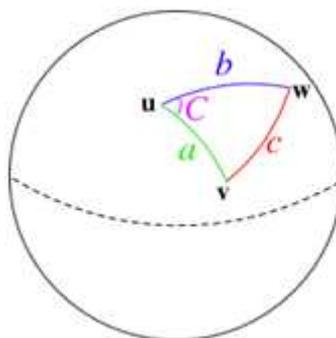


Figure 115 - Sur une sphère, les arêtes d'un triangle sont des arcs - Trigonométrie sphérique

$$d = \text{acos}(\sin(\text{lat}1) * \sin(\text{lat}2) + \cos(\text{lat}1) * \cos(\text{lat}2) * \cos(\text{lon}2 - \text{lon}1)) * R$$

Équation 2 - Distance du grand cercle

$$d = 2 * \text{asin} \left(\sqrt{\sin^2 \left(\frac{\text{lat}2 - \text{lat}1}{2} \right) + \cos(\text{lat}1) * \cos(\text{lat}2) * \sin^2 \left(\frac{\text{lon}2 - \text{lon}1}{2} \right)} \right) * R$$

Équation 3 - Formule d'Haversine

Cependant, l'affirmation qui dit que la « Terre est ronde » est fautive. Elle est en fait plus proche d'un ellipsoïde que d'une sphère, la Terre étant aplatie au niveau des pôles. Ainsi le rayon de la Terre au niveau de l'équateur est de 6378km tandis qu'aux pôles elle n'est que de 6357km (il y a donc 21km de différence). La Terre peut donc être représentée par un ellipsoïde. Un ellipsoïde est défini par deux paramètres : les longueurs du demi petit (r) et du demi grand (R) axe définissant un coefficient d'aplatissement $f = (R-r)/R$. Il est aussi courant de représenter un ellipsoïde par le facteur d'aplatissement et la longueur du demi grand axe. On aimerait que cet ellipsoïde représentant la Terre colle au niveau des mers. Cependant, la forme de la Terre au niveau des mers (représenté par le géoïde) n'est pas symétrique et ne correspond pas à un ellipsoïde (Figure 116).

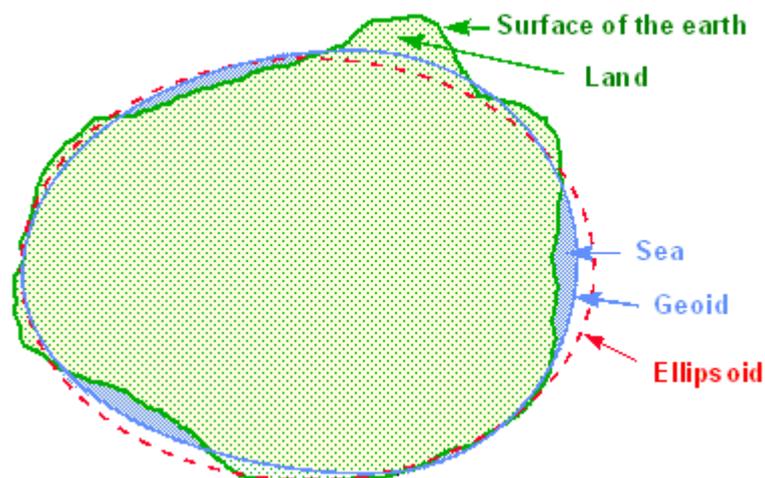


Figure 116- Ellipsoïde et Géoïde

Le géoïde de la Terre étant irrégulier (du fait de la répartition non uniforme de la gravité terrestre), il est trop complexe pour être utilisé directement en cartographie. La Terre est donc représentée en utilisant des ellipsoïdes approximatifs. Ainsi, il est possible d'approximer localement le géoïde par un ellipsoïde local (au niveau d'une région / d'un pays) ou par une ellipsoïde global (Figure 117). Le modèle global retenu actuellement pour une représentation totale du globe est l'ellipsoïde WGS84 (World Geodesic System 1984) qui a été revu en 2004 et dont la validité court jusqu'en 2010 (les ellipsoïdes devant être corrigés avec le temps, dû à différents facteurs comme l'érosion, la dérive des continents,...). En France est utilisé l'ellipsoïde de Clarke 1880 dont le point fondamental se situe au niveau du Panthéon.

Le calcul des distances dans le modèle ellipsoïdal se fait généralement avec la formule de Thaddeus Vincenty (Vincenty, 1975). Cette formule a été récemment validée par des chercheurs australiens qui ont montré qu'elle était précise à 0,115mm (Thomas, et al., 2005).

Ainsi, l'API de la CAORTO propose le calcul des distances via la formule de Vincenty (cf. Annexe A) en utilisant par défaut l'ellipsoïde WGS84, mais propose aussi l'approximation du calcul par la distance des grands cercles et par la formule d'Haversine.

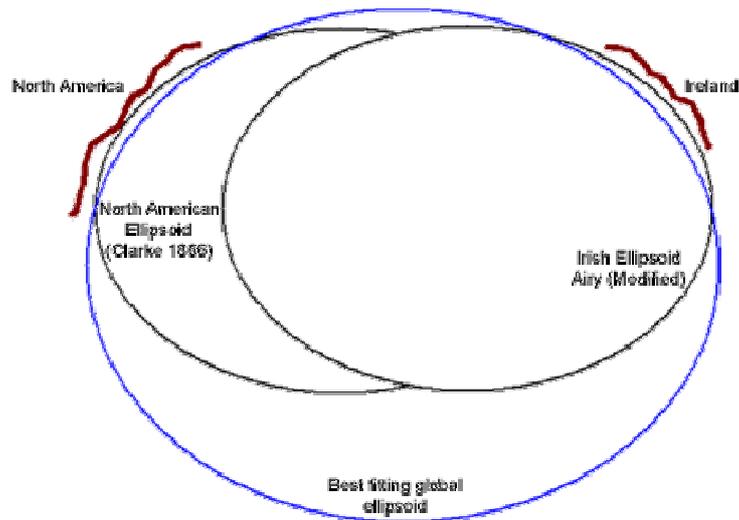


Figure 117 - Ellipsoïdes locales et globale pour représenter la Terre

d) Données administratives

Enfin, il est possible de récupérer les coordonnées des zones administratives gratuitement sur le site Geonames (toujours sous licence Creative Commons). Ces données sont elles aussi intégrées dans la CAORTO et il est donc possible de retrouver les noms du pays, région, ville, arrondissement,... à une position donnée. Cela peut permettre d'établir des règles génériques dans un système ADAS qui ne seraient applicables que dans certaines zones et ou pays. Par exemple, en France la limitation de vitesse sur autoroute est de 110 ou 130km/h, alors qu'en Allemagne il n'y a pas de limitation imposée sur ce type de route.

e) Affichage et projections

D'autres outils sont proposés par l'API qui sont plus d'ordre cosmétique que fonctionnel. Par exemple, il est possible de récupérer une image du graphe routier de la zone en cours et de pouvoir y dessiner facilement des informations directement en coordonnées GPS (et non pixélique), de pouvoir y sur-imprimer des images,... Cela dit, il n'est pas si évident d'afficher une carte correctement : afficher (projeter) une sphère en 3 dimensions sur une surface plane en 2 dimensions.

Il existe plusieurs systèmes de projection pour représenter la Terre, mais une projection ne peut jamais se faire sans qu'il y ait déformation. Pour s'en convaincre, il suffit d'essayer d'aplatir la peau d'une orange. Néanmoins, par le calcul il est possible de définir le type et les paramètres d'une projection dans le but de minimiser certaines déformations. Il est alors possible de :

- soit conserver les surfaces (projections équivalentes)
- soit conserver les angles (projections conformes)
- soit d'opter pour une représentation ne conservant ni les angles ni les surfaces (projections aphyllactiques)

Une projection dite équidistante conserve les distances à partir d'un point donné. Dans tous les cas, aucune projection ne peut conserver toutes les distances.

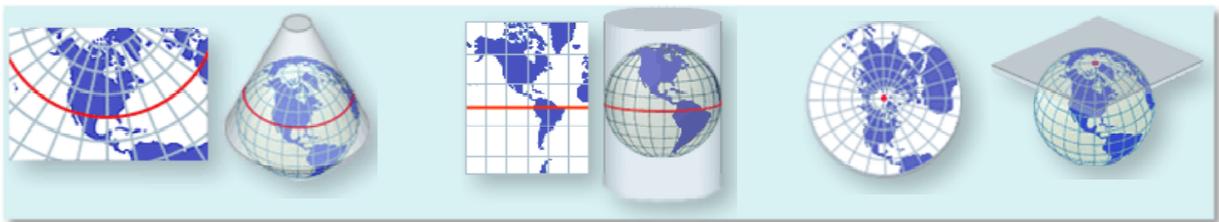


Figure 118 - Projections conformes conique, cylindrique et azimutale

En France, la projection usuellement utilisée est une projection conique (Figure 118) : la projection Lambert. Afin de rendre la projection plus conforme à la réalité (i.e. réduire les déformations), 4 cônes de projection sont utilisés (Figure 119). Une 5^{ème} zone Lambert a été définie, lorsque la cartographie porte sur plusieurs zones et est utilisée pour représenter toute la France : le Lambert 2 étendu.

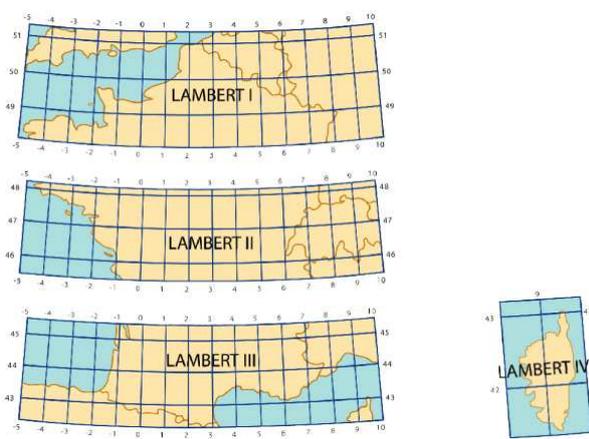


Figure 119 - Projection conique Lambert d'usage en France

La plupart des cartes dans le monde sont affichées avec une projection Mercator (Figure 120) qui est une projection cylindrique directe (Figure 118) facile à calculer (Équation 4). Cette projection

déforme la taille des surfaces qui sont loin de l'équateur. Par exemple, le Groenland est représenté aussi large que l'Afrique alors que l'Afrique est en réalité 14 fois plus grande. Pour pallier ce problème il est possible d'utiliser une projection cylindrique transverse (Figure 118) où il est clair que la déformation sera constante quelque soit l'éloignement à l'équateur. C'est ainsi qu'est défini la projection UTM - Universal Transverse Mercator qui est maintenant plus couramment utilisée. En fait, il n'y a pas 1 mais 60 projections en UTM. En effet, afin que le cylindre soit toujours tangent à la Terre quelque soit la zone cartographiée, la Terre est divisée en 60 zones verticales où en chacune est effectuée une projection cylindrique transverse (Figure 121). Ainsi, trois zones UTM parcourent la France.

Le lecteur désirant plus de renseignement sur les systèmes de projection et de représentation de la Terre pourra se référer à (Robinson, et al., 1995).

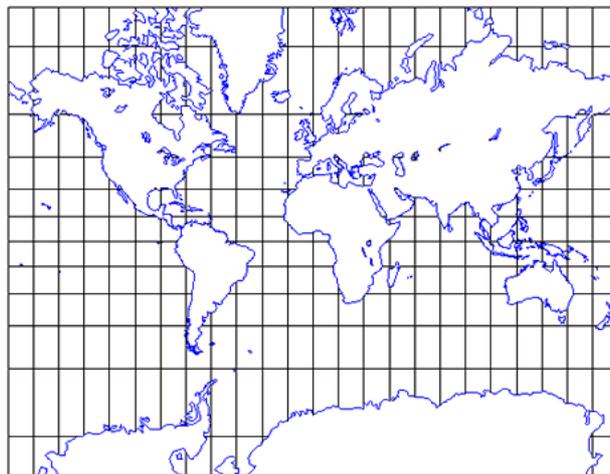


Figure 120 - Projection Mercator – Plus on s'éloigne de l'équateur, plus les aires sont étirées

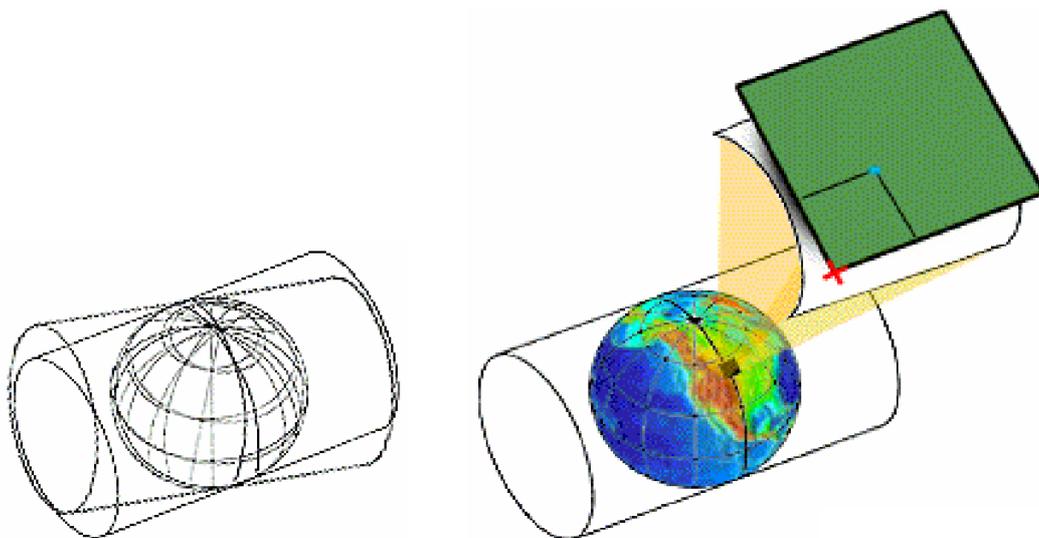


Figure 121 - La projection UTM consiste en plusieurs projections séparées pour chaque méridien tous les 6 degrés

Il est possible au sein de l'éditeur, et ce pour faciliter la création à la main du graphe routier, d'afficher les images satellitaire de Google. Google utilise pour ses cartes le système de projection Mercator (Équation 4). C'est pourquoi, l'API et l'éditeur de la CAORTO utilisent ce système de projection afin de rendre plus « réaliste » (i.e. conforme à ce que nous avons l'habitude de voir en représentation de carte) l'affichage de la cartographie et de ne pas déformer les images satellitaire.

$$x = lon$$

$$y = \ln\left(\tan(lat) + \frac{1}{\cos(lat)}\right)$$

Équation 4 - Projection Mercator

2.5. Intégration et résultats de travaux futurs

Afin de valider l'implémentation des algorithmes de calcul des distances de la CAORTO, ceux-ci ont été comparés avec ceux développés par des systèmes commerciaux. Ainsi, deux points GPS ont été relevés sur Google Maps, la distance entre ces deux points a été calculée avec Google Maps et avec FizzyCalc ; puis avec l'algorithme de Vincenty en utilisant l'ellipsoïde WGS84, le formule d'Haversine et le calcul des grands cercles de la CAORTO. L'API de la CAORTO permet aussi de convertir les coordonnées GPS en coordonnées dans le système de projection Lambert 2 étendu (utilisé en France) qui fournit des coordonnées dans un repère cartésien en mètre. La conversion des deux points GPS référence ici en Lambert permet ainsi de calculer directement leur distance. Le Tableau 14 montre que tous ces calculs sont conformes à 30 cm près, ce qui gage d'une bonne implémentation et de bons résultats pour la CAORTO.

	Latitude	Longitude
Point A	48.845579	2.339834
Point B	48.843855	2.338878

Logiciel	Distance (m)	Formule (CAORTO)	Distance (m)
Google Maps	204.29	Lambert2e	204.26
FizzyCalc (Vincenty WGS84)	204.15	Vincenty WGS84	204.15
FizzyCalc (Haversine)	203.92	Haversine	204.06
FizzyCalc (Grand Cercle)	203.92	Grand Cercle	204.06

Tableau 14 - Comparaison du calcul des distances des Points A et B avec la CAORTO et d'autres logiciels

En ce qui concerne la procédure de map-matching, comme prévu du fait de l'utilisation d'un DGPS assez précis (d'une précision théorique d'1 mètre), le calcul du segment le plus proche au segment en cours utilisé dans la CAORTO, ne pose de problème qu'au niveau des intersections. Ainsi sur un carrefour par exemple, la notion de segment le plus proche est trop ambiguë et peut très bien se retrouver être n'importe lequel des segments joignant le carrefour. Par exemple, Figure 122, si le GPS se « trompe » et retourne la position rouge, alors le segment le plus proche sera le A, si c'est la position bleue qui est fournie, alors le segment le plus proche calculé sera le C, ... n'importe quelle route à ce carrefour peut être le segment le plus proche quand le véhicule s'y trouve au centre. Avec un GPS classique (d'une imprécision de 10 mètres), ce phénomène serait encore plus grand et l'erreur de map-matching, qui est ici très courte, du fait qu'un véhicule ne reste généralement pas longtemps sur un carrefour, s'en ressentirait fortement. Il serait bon, dans des travaux futurs d'incorporer un processus de map-matching beaucoup plus robuste fusionnant des informations proprioceptives (comme l'odométrie, l'angle au volant,...) afin de savoir si le véhicule tourne et sélectionner le bon segment.

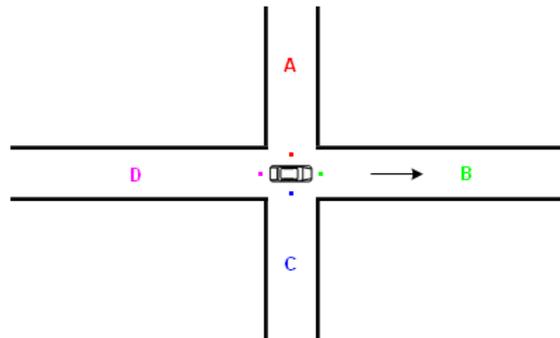


Figure 122 - Imprécision du GPS et choix du segment le plus proche (map-matching)

L'intégration de l'API au sein de notre logiciel de prototypage (^{rt}Maps) est aisée et ne pose guère de problème. Ainsi, comme le montre la Figure 123, l'utilisation des coordonnées GPS permet bien d'afficher la cartographie de l'endroit où le véhicule se trouve. Tout d'abord les noms de la rue, de la ville, du département, de la région et du pays sont corrects. L'information administrative semble donc être correctement sauvegardée et retrouvée. Les fonctionnalités de dessins offertes par la CAORTO permettent ensuite de visualiser et de valider ses diverses fonctionnalités. Ainsi l'affichage de la limitation de vitesse en cours attribuée au segment courant (sous forme d'une image en surimpression du graphe routier), l'affichage en rouge du segment sélectionné par le map-matching, de la position GPS en verte, de la position rétroprojetée en bleue et de la position du prochain carrefour en violet, permet de valider les calculs mis en jeu par ces diverses fonctionnalités. L'utilisation d'une caméra valide les résultats de map-matching et de projection de la position GPS en comparant la position visuelle (par rapport aux bâtiments) de la position cartographique. Enfin, cette validation « visuelle » a eu lieu sur plusieurs séquences dans Paris, et n'a pas révélé de problème particulier (hormis le map-matching aux carrefours). Elle a en outre prouvé le bon comportement en temps réel de la CAORTO (pour la récupération des données routières dans la base et l'affichage de ces informations).

En ce qui concerne l'éditeur de donnée, c'est celui-ci qui a permis d'importer (et de traduire) les données d'OpenStreetMap et de GeoNames. Au vu des précédents résultats, il est clair que cette importation fonctionne et sauvegarde correctement dans le bon format les données dans la base de la CAORTO. L'affichage et l'édition des données dans une même interface (Figure 124) permet aussi de valider ces deux fonctionnalités : en effet, la sélection d'un segment permet d'afficher ses attributs, la modification de l'un deux, comme le sens de circulation entraîne la sauvegarde de la nouvelle donnée et un changement d'affichage (après relecture des données). Enfin, l'affichage en fond des images satellitaires Google permet de valider l'affichage en projection Mercator du graphe routier ainsi que les coordonnées GPS (i.e. la position) des routes (Figure 125).

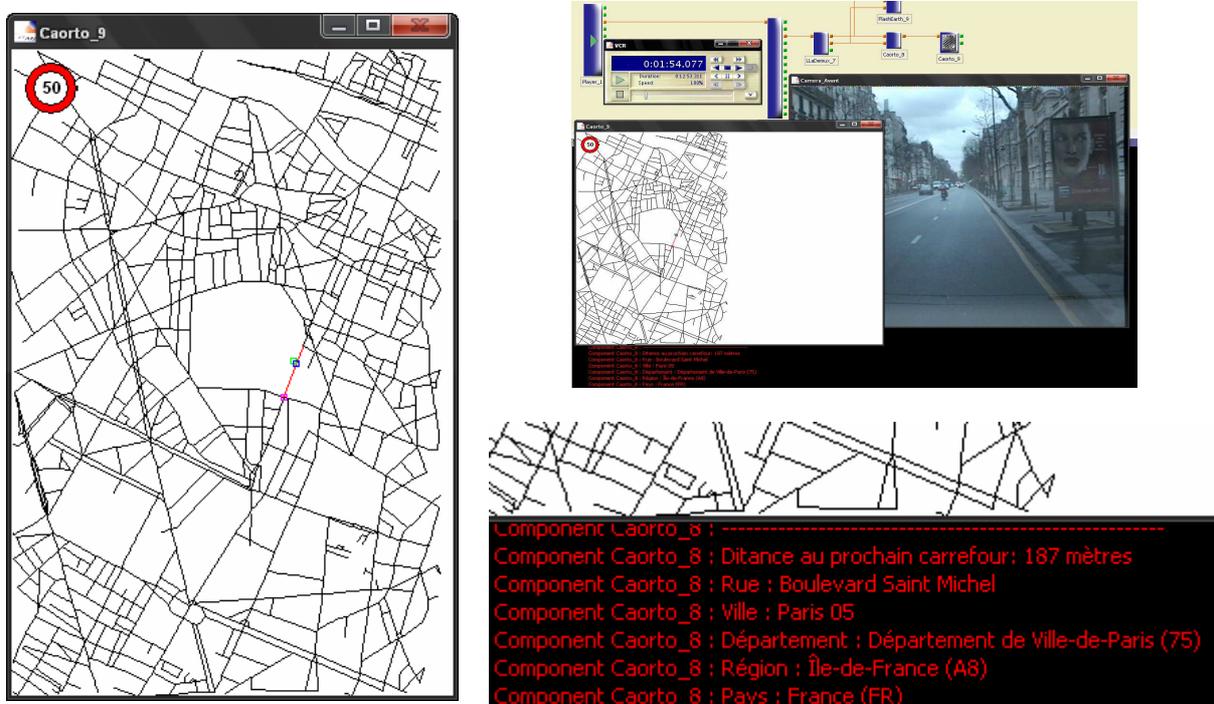


Figure 123 - Intégration de la CAORTO dans rtMaps (à gauche le graphe routier avec diverses informations projetées, en haut à droite une vue d'ensemble de l'intégration, en bas à droite un zoom sur l'affichage des informations administratives)

L'une des équipes du laboratoire a mis au point une technique de numérisation et modélisation 3D d'environnements extérieurs, utilisant un système mobile appelé LARA-3D. Il s'agit d'une voiture équipée d'un système de localisation géographique précis (GPS, Centrale Inertielle), d'un télémètre laser fixé à l'arrière du véhicule, et de caméras. Ce dispositif, utilisé en environnement urbain et routier, permet de recueillir des nuages de points 3D décrivant avec une bonne précision les éléments présents le long des trajets effectués (routes, ronds-points, façades, arbres, voitures...), et

après traitement d'avoir des modèles par facettes des scènes numérisées, ainsi que des couleurs sur les points et des textures photo-réalistes sur les facettes (Brun, 2007).

Une des idées émises étaient de pouvoir sauvegarder ces informations 3D. Or, extraire les informations (largeur de la route, rayon de courbure vertical,...) d'un modèle 3D en vue d'être intégrées dans une base cartographique et surtout utilisées n'est pas chose aisée et n'est pas encore d'actualité bien que des travaux sont en cours sur ces problématiques. Cependant, les modèles 3D issus de ces scans sont géo-référencés. C'est pourquoi, dans un premier temps, ces modèles bruts ont été intégrés afin de pouvoir être affichés les uns à côté des autres sur une carte, telle la CAORTO (Figure 126). Cet affichage ne constitue pour l'instant qu'une « démonstration technologique » et ne prend pas en compte par exemple l'élévation du terrain. Nous n'avons pas encore réfléchi sur la façon de stocker intelligemment les modèles 3D (s'il faut par exemple les segmenter par zone et les stocker de façon hiérarchique ou non, comme pour le graphe routier,...) ?

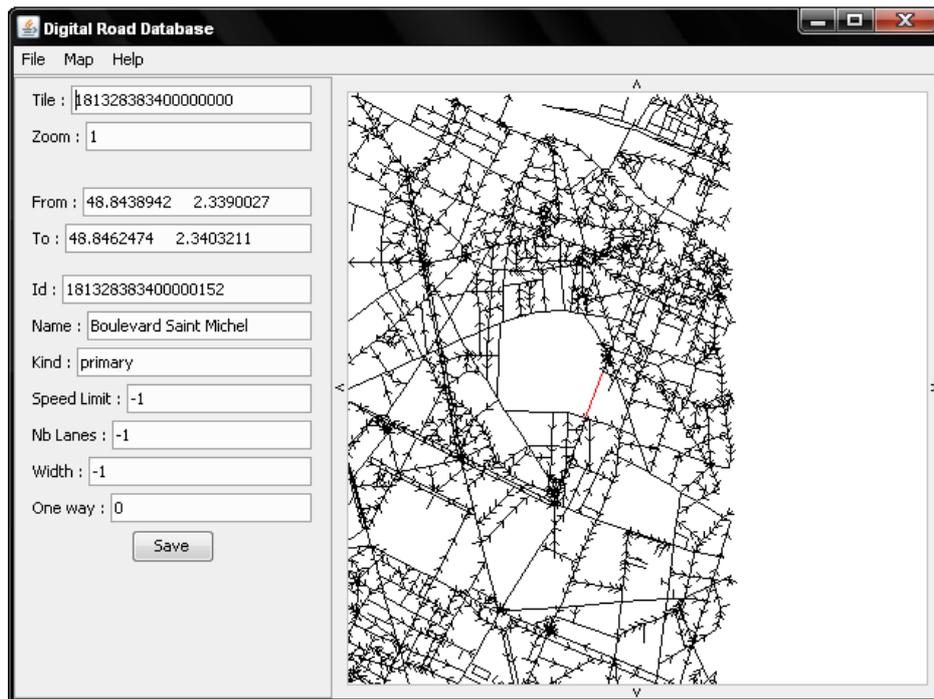


Figure 124 - L'éditeur de la CAORTO permet l'affichage et l'édition des données

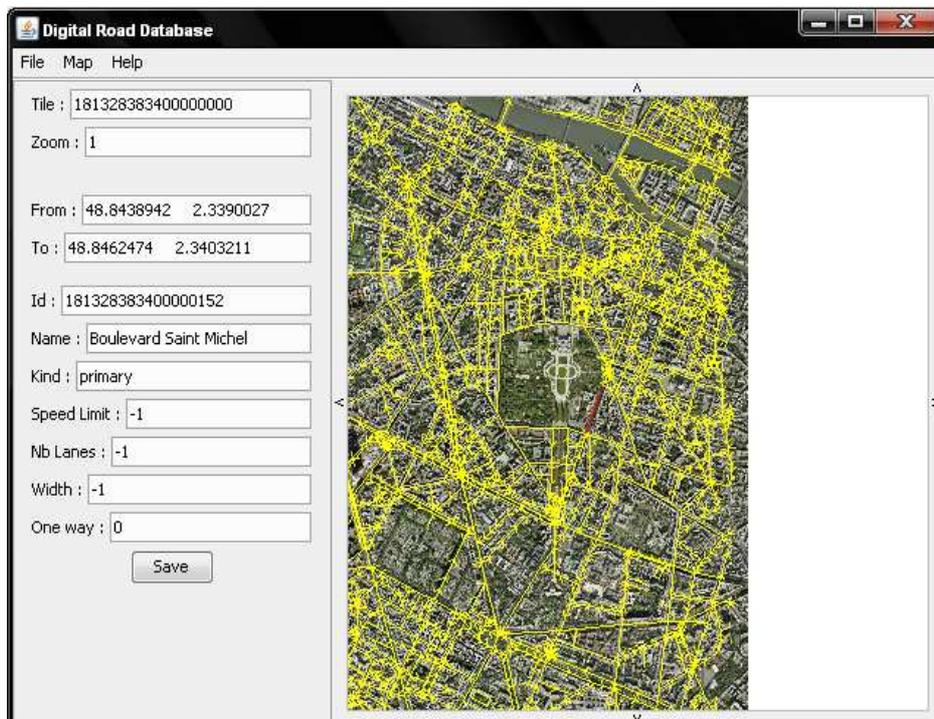


Figure 125 - L'éditeur de la CAORTO avec affichage des images satellitaire Google

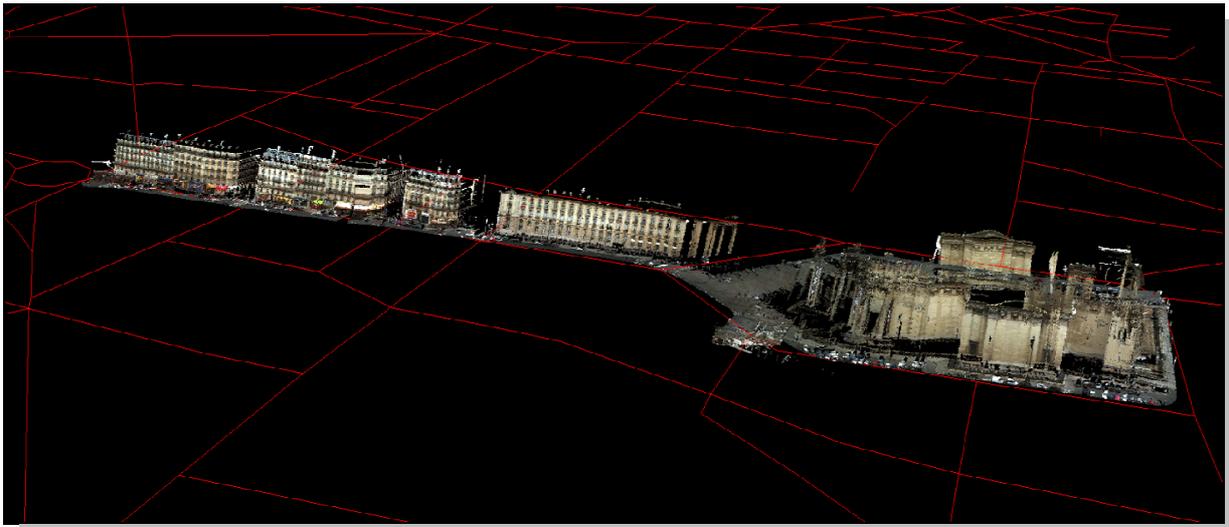


Figure 126 - Intégration et affichage 3D de 3 modèles 3D juxtaposés sur la carte (avec ou sans affichage des données satellitaires Google)

Enfin la CAORTO est déjà utilisée dans d'autres projets du laboratoire comme :

- SLA en collaboration avec Valeo, qui vise à effectuer une détection intelligente de la limitation de vitesse
- SpeedCam, pour la création d'un ACC (Advanced Cruise Control) légal et intelligent
- DIVAS, pour le stockage de données de l'infrastructure (ex : l'adhérence des routes,...)

La CAORTO sera aussi sans doute utilisée dans les travaux de thèse futurs du laboratoire, notamment les travaux ADAS où il sera possible par exemple de déclencher la détection des feux tricolores aux approches des carrefours et d'orienter les zones de recherche dans l'image.

Enfin, il n'est pas exclu d'utiliser nos divers algorithmes de détection visuelle (ex : détection de panneaux de limitation de vitesse, des feux tricolores, ...) afin d'enrichir la CAORTO en y enregistrant de façon automatique et en temps réel ses « attributs » de route.

L'utilisation de la CAORTO peut être aisément exploitée afin de connaître la limitation en cours (Figure 123) au cas où le système visuel n'ait pas encore détecté et reconnu de panneau de limitation de vitesse. Cependant, dans le cas schématisé Figure 127, la position GPS qui souffre d'une certaine imprécision (10 mètre en cas normal, cercle bleu sur la figure), fait qu'il n'est pas possible de savoir sur quelle voie le véhicule se trouve (sur l'autoroute ou sur la voie de sortie). Le capteur cartographique seul ne peut donc pas déterminer la limitation de vitesse même si celle-ci est renseignée comme attribut des segments. Or, le système de reconnaissance visuel peut très bien détecter les 2 panneaux de vitesse (70 et 110 de la voie de sortie et de l'autoroute) qui sont à proximité et être lui aussi dans l'incapacité de prendre une décision quant à la limitation de vitesse en cours ! L'utilisation de la théorie de Dempster Shafer (comme il est fait dans l'état de l'art) ne résoudrait rien. En effet, les 2 systèmes renvoyant chacun 70 et 110 en guise d'éléments focaux (de détection), il serait impossible de fusionner ces résultats pour déterminer la limitation qui est la bonne sans renseignement extérieur (la voiture pouvant très bien être positionnée sur la voie de sortie).

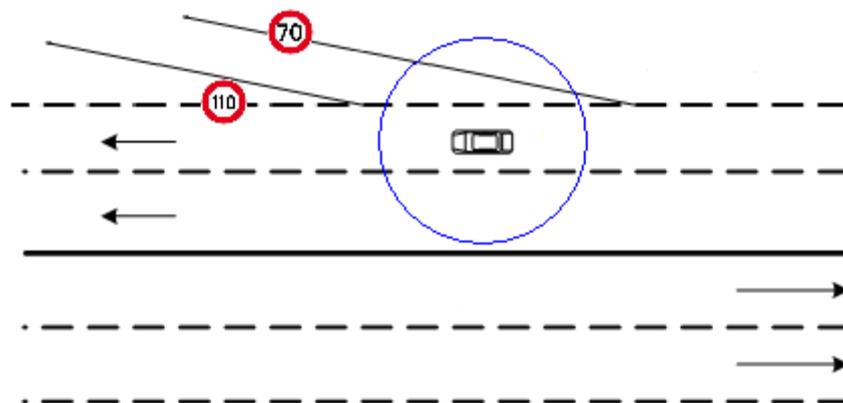


Figure 127 - Cas où l'utilisation d'un capteur cartographique seul ne suffit pas à déterminer la limitation de vitesse (ellipse d'incertitude de positionnement GPS en bleu)

Afin de résoudre ce problème, on se propose d'utiliser l'information issue de la détection des lignes de marquage au sol décrite dans le chapitre suivant.