



L3.3 Procédures d'enrichissement

Work Package	3
Responsable du WP, affiliation	Tattegrain Hélène Lescot
Livrable n°	L3.3
Version	VF
Auteur responsable du livrable	THIOLON Gildas, Ceesar
Auteurs, affiliations	Gildas THIOLON, CEESAR Claire NAUDE, LMA Hélène Tattegrain, LESCOT
Relecteurs, affiliations	MIGNOT Dominique
Statut du livrable	Final

Veillez citer ce document de cette façon :

Thiolon, G., Naude, C., & Tattegrain, H. (2022). *Livrable L3.3 : Procédure d'enrichissement*. Projet SURCA. 31 p.

Historique des révisions

Version	Date	Auteurs	Type des changements
V0	07/03/2022	Gildas THIOLON	Définition du plan
V1	29/06/2022	Claire NAUDE	Intégration des enrichissements effectués dans DYMOA
V2	05/08/2022	Gildas THIOLON	Intégration des enrichissements effectués dans MOOVE & UDRIVE
V3	31/08/2022	Clément VAL	Relecture/Corrections
V4	31/08/2022	Hélène Tattegrain	Intégration des enrichissements effectués dans UDRIVE pour l'analyse de l'influence des piétons sur la vitesse moyenne

Remerciements

Le Projet SURCA est financé par la dévolution de la Fondation Sécurité Routière, la Délégation à la sécurité routière et pour moitié par les partenaires du projet.



Résumé du projet Surca

Les questions posées par la cohabitation de véhicules de plus en plus automatisés avec des véhicules conventionnels et des usagers vulnérables, cyclistes, piétons, deux-roues motorisés, sont au cœur des préoccupations des décideurs publics, constructeurs, ou spécialistes de l'infrastructure routière et de la sécurité routière. Tous ont l'espoir que ces nouvelles technologies contribuent à améliorer la sécurité routière. L'objectif global du projet « Sécurité des Usagers de la Route et Conduite Automatisées, SURCA » est de contribuer à une meilleure intégration de la Conduite Automatisée dans la circulation actuelle. Les partenaires du projet (Ifsttar, DSR, Ceesar, Cerema, Vedecom, Lab), ont ainsi comme objectif d'identifier quelles interactions existent et quelles stratégies pertinentes sont mises en place par les conducteurs pour proposer des recommandations aux concepteurs de véhicules autonomes sur les besoins en termes d'interactions et en termes de comportement du véhicule autonome. Pour cela, il est prévu d'analyser des bases de données existantes sur la conduite des véhicules conventionnels et d'identifier les facteurs qui peuvent expliquer des comportements différents.

Les connaissances issues de ces bases seront utilisables pour simuler l'introduction de la conduite automatisée de niveaux 3, 4 et 5, avec des taux de pénétration faibles. La gestion des interactions avec les autres usagers doit être réalisée dès que le véhicule peut évoluer en autonomie sans supervision du conducteur, quelles que soient la durée et les sections sur lesquelles cette automatisation sera possible. En cas de taux de pénétration très important, d'autres types d'interactions risquent de se mettre en place et devront alors être étudiés.

Ce projet est articulé autour de deux sous-thématiques :

- L'identification des scénarios d'interaction entre véhicules autonomes et autres usagers de la route (véhicules conventionnels, deux roues motorisés, cyclistes, piétons), avec un focus particulier sur les personnes âgées :
 - Etude des situations de négociation où les conducteurs gèrent cette interaction humaine, à partir de bases de données de conduite conventionnelle, et en utilisant des éléments difficilement émis et perçus par les systèmes automatisés (regard, connaissance a priori d'intention, etc.),
 - Etude de la réaction des autres usagers face à un véhicule autonome alors que son conducteur est absorbé dans une tâche annexe,
 - Identification des besoins de communication du véhicule autonome en phase active avec les autres usagers,
 - Analyse des besoins des usagers âgés et acceptabilité sociétale du véhicule autonome.
- L'étude des impacts de la posture des occupants (conducteur et passagers) d'un véhicule en mode autonome sur le risque lésionnel :
 - Choix des scénarios de simulation : positions des occupants, conditions de choc (lors de la réalisation de tâches annexes) et systèmes de retenue,
 - Evaluation des lésions potentielles par simulations numériques en fonction des systèmes de retenue (par ex. déploiement d'air bag),
 - Recommandations en termes de postures acceptables selon les différents systèmes de retenue.

Résumé Livrable 3.3

L'objectif de ce document est de présenter les enrichissements automatiques apportés, lorsque nécessaire, aux différentes bases de données présentées dans le livrable 3.1. Ces enrichissements concernent la contextualisation de données continues, la sélection de séquences pertinentes parmi une base de données étendue, et le calcul d'attributs nécessaires à la caractérisation de ces scènes, sur lesquelles porteront les études effectuées dans les WP4, WP5 et WP6. Ce document est à destination des évaluateurs et doit permettre d'identifier la source des données disponibles pour analyse, en présentant pour chaque élément :

- Une description du fonctionnement de l'algorithme utilisé pour calculer le signal/ attribut
- Les sorties de l'algorithme, en précisant bien pour chaque élément, son unité et les valeurs possibles pouvant être prises par le signal/attribution
- Eventuellement les limites identifiées de l'algorithme

Les bases de données concernées par ces enrichissements sont UDRIVE, MOOVE et DYMOA, qui ont été présentées dans le livrable 3.1

Table des matières

1	Introduction	7
2	Base UDRIVE.....	9
2.1	Calculs de signaux continus	10
2.1.1	Nombre de piétons	10
2.1.2	Changement de voie.....	10
2.1.3	Détection de Virage	11
2.1.4	Détection de Voies à Chaussées Séparées (VCS)	11
2.1.5	Cut-in	13
2.1.6	Appels de phares	13
2.2	Détections de scènes pertinentes	14
2.2.1	Cut-in sur VCS en section courante	14
2.2.2	Cut-in sur VCS en situation de suivi	14
2.2.3	Sections courantes pour l'analyse de l'impact de la présence de piéton	15
2.2.4	Intersections pour l'analyse de l'impact de la présence de piéton	16
2.3	Caractérisation des scénarios	17
2.3.1	Cut-in sur VCS en suivi	17
2.3.2	Caractérisation des Intersections et sections courantes pour l'analyse de l'impact de la présence de piéton	23
3	Base MOOVE.....	26
3.1	Calculs de signaux continus	26
3.1.1	Fusion « Haut Niveau ».....	26
3.1.2	Cut-in	27
3.1.3	Remontée de file	28
3.2	Détections de scènes pertinentes	29
3.2.1	Type d'obstacles	29
3.3	Caractérisation des Scénarios.....	29
3.3.1	Scénario remontée de file	29
3.3.2	Scénario Cut-in	30
4	Base DYMOA	32
4.1	Rappel sur les données DYMOA.....	32
4.2	Caractérisation des Scénarios	32
4.2.1	Reconstruction des trajectoires.....	32
4.2.2	Simulation détection du 2RM	34
5	Annexe	35
5.1	Glossaire des obstacles.....	35
5.2	Type de données en sortie des enrichissements	35

Tables des illustrations

Figure 1 : Schématisation du processus Collecte-Enrichissement-Analyse	7
Figure 2: Capture d'écran de l'outil SALSAS.....	9
Figure 3 : Exemple de Changement de voie à Gauche puis à Droite, détectés avec les distances au marquage.....	10
Figure 4 : Schéma orientation véhicule	11
Figure 5 : Exemple de Cut-in avec discontinuité et réduction de la distance à la cible ET Absence de changement de voie de l'égo.....	13
Figure 6 : Schéma d'un Cut-in avec Cible.....	14
Figure 7 : Répartition des positions des cibles détectées sur VCS	15
Figure 8 : Illustration des sections courantes	16
Figure 9 : Illustration des intersections.....	16
Figure 10 : Diminution des distances de sécurité pendant un Cut-in avec cible.....	17
Figure 11 : Exemple de détection de débuts de manœuvre de Cut-in :	18
Figure 12 : Illustration du calcul de TTC entre l'égo et le cutter	21
Figure 13 : Phases temporelles de calcul des caractéristiques sur les intersections	23
Figure 14 : Positions et angles de vue des capteurs extéroceptifs des véhicules MOOVE.....	26
Figure 15 : Définition des instants T1 à T6 pour lesquels les données ont été extraites.....	29
Figure 16 - Définition des instants T1, T2, T3 et T4.....	30
Figure 17 : Trajectoire du 2RM repositionnée sur une carte à l'échelle	33
Figure 18 : Format des données dans Excel (temps, distance, vitesse, décélération, coordonnées GPS, angle de lacet, VL puis 2RM, longueur et largeur)	33
Figure 19 : Trajectoires des deux véhicules en interaction	33
Figure 20 : Evolution avant détection et détection	34

1 Introduction

Les bases de données concernées par l'enrichissement sont des bases où l'on retrouve des données de conduites enregistrées en continu et parmi lesquelles il faut pouvoir identifier et caractériser les scénarios identifiés comme pertinents pour le projet SURCA. Les données de roulage constituant ces bases sont collectées au sein de système embarqués, elles ont une précision et une qualité dépendant du système et des capteurs utilisés.

Les données ainsi collectées dans ces bases sont « **Brutes** », c'est-à-dire qu'elles sont directement fournies par les capteurs après perception et acquisition. En fonction de la qualité de ces données un filtrage/ nettoyage des données peut être nécessaire, avant d'appliquer des « règles haut-niveau » permettant de détecter, puis de caractériser les scénarios voulus. Ces différentes étapes constituent l'enrichissement décrit dans ce livrable

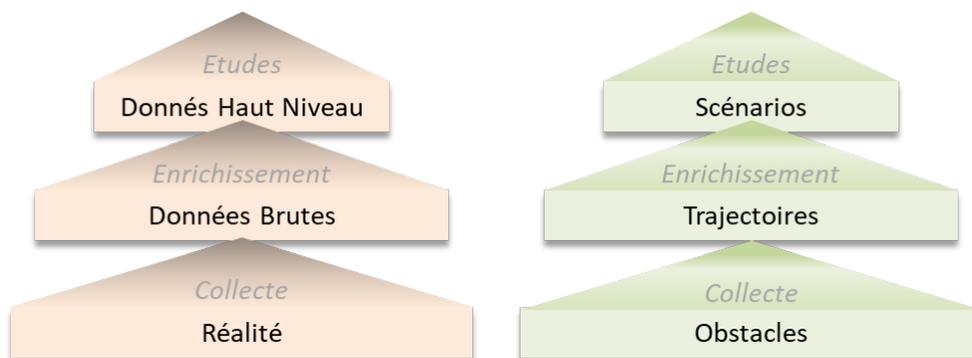


Figure 1 : Schématisation du processus Collecte-Enrichissement-Analyse

Par exemple, les obstacles évoluant autour du véhicule de collecte font partie de la réalité. Les capteurs (radars, lidars, caméra...) vont permettre d'enregistrer les positions relatives à l'égo de chaque obstacle et à chaque instant. Ces positions correspondent aux données brutes. Un enrichissement spécialement mis au point pour le système de collecte va détecter des trajectoires particulières, par exemple un coup de frein du véhicule devant l'égo, ce qui va permettre d'identifier les scénarios de ce type dans la base de données. La qualification de ces trajectoires particulières correspond, elle, aux données haut-niveau. Ces données haut-niveau sont ensuite consommées par les WP 4, 5 et 6.

Les enrichissements produits doivent s'adapter aux données d'entrée pour obtenir les meilleures performances, les algorithmes ainsi mis au point sont donc spécifiques à chaque base de données et seront décrits de façon indépendante pour chaque base (UDRIVE, MOOVE...).

Les enrichissements effectués sur chaque base de données pour permettre les analyses peuvent être distingués en 3 catégories :

1. **Calcul de signaux continus.** Il s'agit d'une valeur calculée à chaque instant à partir des données brutes. (Exemple : Type de route) ;
2. **Détection de scènes pertinentes.** Il s'agit de signaux détectant certains événements relativement rares qui interviennent pendant la collecte de données. Cela intègre également le tri effectué pour sélectionner certaines scènes détectées (Exemple : Remontée de file faisant intervenir un 2RM uniquement) ;
3. **Caractérisation des scénarios identifiés.** En plus des attributs nécessaires à la détection des scénarios, l'analyse peut demander le calcul d'indicateurs qui permettront de qualifier ce scénario. Exemple : le temps inter véhicule pour les scénarios d'insertions devant l'égo.

Pour chaque enrichissement, une description du fonctionnement et des sorties de l'algorithme sont présentées. Cela comprend notamment les différents types de données, qui sont définis en annexe de ce document. Par ailleurs un rapide glossaire décrivant les *obstacles* acteurs des différents scénarios est

également présent en annexe. Ces obstacles sont mentionnés en *italique* dans la description des algorithmes.

2 Base UDRIVE

Pour rappel, la base UDRIVE est une base de données en conduite naturelle en Europe, enregistrée entre 2015 et 2017 dans plusieurs pays européens. Cette base de données contient notamment les données collectées par 122 véhicules légers de particuliers (186 conducteurs) recrutés parmi le grand public. L'instrumentation réalisait l'enregistrement des données suivantes :

- **CAN du véhicule** : variables dynamiques, utilisation des commandes et des fonctions du véhicule par le conducteur ;
- **GNSS** : géolocalisation ;
- **Centrale inertielle** : accélérations, vitesses angulaires et compas sur 3 axes ;
- **Caméra intelligente** : détection des autres usagers et de leur trajectoire, lecture des panneaux routiers, position du véhicule sur sa voie ;
- **Caméras vidéo de contexte** : Vues extérieures avant du véhicule et comportement du conducteur (de la conductrice) à l'intérieur.

L'enregistrement est continu lors de l'ensemble des roulages des véhicules, et chaque véhicule restait instrumenté pour une durée d'environ 18 mois. Ainsi, une base de données de 2 millions de kilomètres représentant une large variété de conditions d'usage a été constituée.

En France, 43 conducteurs utilisant 30 véhicules différents, recrutés dans la région Lyonnaise, ont permis la collecte de **12 600 heures de conduite / 573 000 km**, correspondant à l'ensemble de leurs trajets quotidiens, professionnels, de loisir... Les roulages français ont commencé fin 2015, et se sont terminés mi 2017. Ces trajets totalisent 49832 enregistrements.

L'analyse et l'enrichissement des données a été réalisée sur le logiciel SALSA, développement interne du CEESAR. SALSA permet de requêter et visualiser les données à la demande, mais aussi de développer et appliquer des algorithmes sur l'ensemble de la base de données, ainsi que d'annoter, manuellement ou automatiquement, les données.

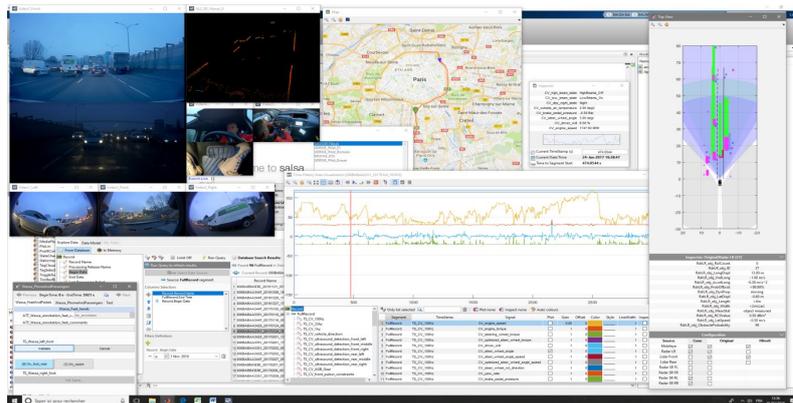


Figure 2: Capture d'écran de l'outil SALSA

SALSA permet de développer de manière interactive les algorithmes avec l'ensemble des visualisations, d'exécuter de manière contrôlée les algorithmes sur l'ensemble de la base, et d'identifier et caractériser aisément des événements particuliers (les « aiguilles dans une botte de foin ») correspondant aux scénarios du projet SURCA.

Par ailleurs pour les besoins du projet UDRIVE (ainsi que de nombreux projets ultérieurs utilisant la base UDRIVE) plusieurs enrichissements ont été développés et ont pu être réutilisés pour le projet SURCA. Cela comprend notamment la contextualisation géographique des données consistant en

l'interrogation d'un Système d'Information Géographique (SIG) afin d'obtenir des données supplémentaires sur l'infrastructure routière sur laquelle l'égo évolue (Vitesse Maximale Autorisée (VMA), nombres de voies de circulation).

2.1 . Calculs de signaux continus

2.1.1 Nombre de piétons

- Description :
 - Compter le nombre de piétons détectés dans l'environnement de l'égo. Calculé à partir du type des *obstacles* fourni par le capteur disponible (Caméra intelligente)
- Sortie :
 - Nombre de piétons détectés à chaque instant (uint8)
 - Présence ou non d'au moins un piéton détecté (bool)

2.1.2 Changement de voie

- Description
 - Détection d'un changement de voie de l'égo. Déterminé à partir de l'analyse des distances aux marquages droits et gauches fournis par la caméra intelligente. L'algorithme détecte 2 discontinuités concomitantes des distances à droite et gauche afin de s'assurer que l'égo a changé de voie. Le signe de cette discontinuité permet de distinguer les changements à gauche et à droite.

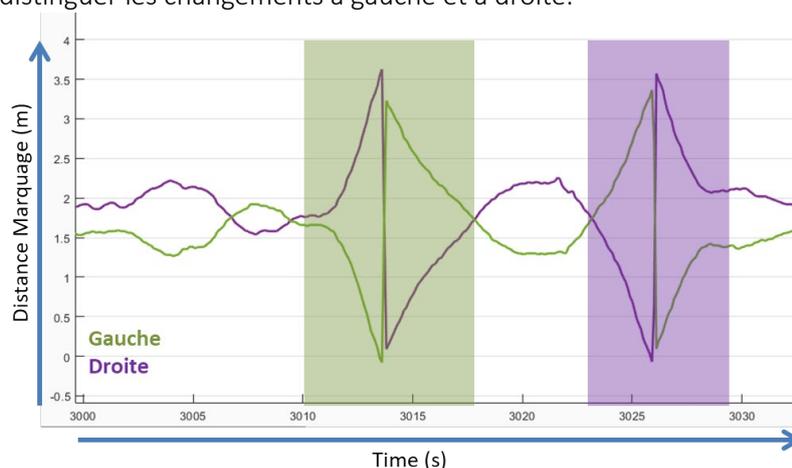


Figure 3 : Exemple de Changement de voie à Gauche puis à Droite, détectés avec les distances au marquage

- Limites
 - La détection de changement dépend directement de la qualité de détection des marquages. Celle-ci est optimisée pour les routes à Voies à Chaussées Séparées (VCS) ou les routes rurales mais donnent de moins bons résultats en contexte urbain. En conséquence les changements de voie sont moins bien détectés.
 - Ne détecte pas les changements de voie vers une voie de largeur rapidement variable (ex : voie de sortie).
- Sortie

- Signal Enumération changement de voie échantillonné à 10Hz qui prend les valeurs suivantes :
 - Pas de changement de voie
 - Changement de voie à gauche
 - Changement de voie à droite

2.1.3 Détection de Virage

- Description
 - Les mesures dynamiques embarquées dans l'égo-véhicule permettent de caractériser sa trajectoire qui peut être associée à celle de l'infrastructure dans le cas de **virages**. On considère qu'un virage existe lorsque l'accélération latérale est maintenue à 1m.s^{-2} au moins, pendant une période qui voit l'orientation du véhicule changer d'au moins 30° . L'accélération latérale est obtenue à partir de la vitesse longitudinale et de la vitesse de lacet (formule de Frenet) ; la variation d'orientation est obtenue par intégration de la vitesse de lacet.

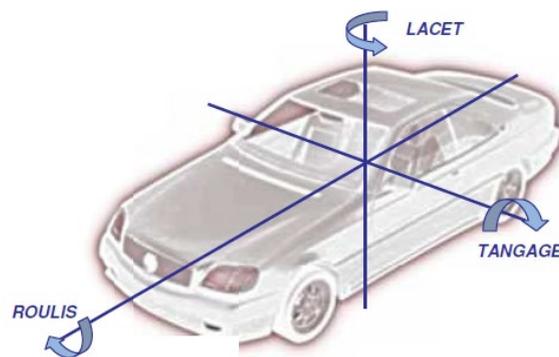


Figure 4 : Schéma orientation véhicule

- Sortie:
 - Présence ou non d'un virage (Booléen). Le front montant correspond au début de la fenêtre durant laquelle l'accélération latérale dépasse 1m.s^{-2} . Le front descendant correspond à la fin de cette même fenêtre.

2.1.4 Détection de Voies à Chaussées Séparées (VCS)

- Description :
 - Détermination du type de chaussées sur laquelle évolue l'égo. On considère que l'égo circule sur une VCS lorsque les conditions suivantes sont respectées :
 1. La Vitesse Maximale Autorisée (VMA) est strictement supérieure à 50km/h
 2. Le nombre de voies de circulation dans le sens de circulation de l'égo est supérieur ou égale à 2
 3. La circulation est à sens unique (Assimilable à la présence d'un séparateur physique entre les 2 sens de circulation)
- Limites
 - L'interrogation d'un SIG présente deux limites intrinsèques :
 1. L'égo doit être positionné sur la bonne chaussée. Cela est plus difficile lorsque la position de l'égo n'est pas fiable (embouchure d'un tunnel, canyon urbain) ou que l'infrastructure est complexe (passage sous un pont, contre-allée) ...

2. Les informations du SIG sur le tronçon identifié sont correctes et à jour.

- Sortie
 - L'*Ego* évolue ou non sur une Voie à Chaussées Séparées (Bool). Ce signal est synchronisé avec la localisation de l'*égo* par GNSS (1Hz)

2.1.5 Cut-in

- Description
 - Présence d'un véhicule s'insérant dans la voie de l'égo devant l'égo. Le véhicule s'insérant est défini comme étant le *cutter*. A la suite de la manœuvre le *cutter* est la nouvelle *cible* (Objet dans la voie de l'Ego, devant l'Ego). Un nouveau Cut-in est détecté quand les conditions suivantes sont réunies :
 - Changement de *cible*
 - ET Discontinuité et Réduction de de l'inter distance entre l'égo et la *cible*

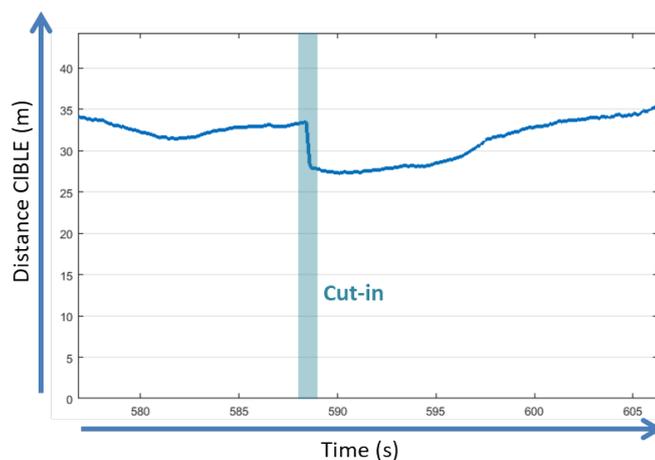


Figure 5 : Exemple de Cut-in avec discontinuité et réduction de la distance à la cible ET Absence de changement de voie de l'égo

- Sorties
 - Signal Booléen événementiel (Echantillonné de manière non uniforme) prenant la valeur VRAIE (front montant) au moment du changement de *cibles*. La valeur FAUX (front descendant) intervient 0.2s après le front montant.
 - Id du *cutter*. Identifiant unique de l'*obstacle* réalisant la manœuvre (int)

2.1.6 Appels de phares

- Description
 - L'utilisation des feux de route est enregistrée. Ceux-ci sont normalement utilisés la nuit, hors agglomération, sans éclairage et sans présence d'autres usagers. Ils peuvent également être actionnés de façon éphémère pour signaler sa présence quel que soit le contexte de conduite. C'est cette utilisation que l'on détecte en vérifiant que la durée d'utilisation ne dépasse pas 0.5s.
- Limite
 - Un appel de phare peut éventuellement durer plus de 0.5s, auquel cas il ne sera pas détecté par cet algorithme ;
 - Un appel de phare peut être utilisé pour exprimer son mécontentement et ne pas seulement signaler sa présence.
- Sortie

- Présence ou non d'un appel de phare (Bool)
C'est un signal évènementiel (une valeur à chaque changement d'état, pas d'échantillonnage à fréquence fixe).

2.2 Détections de scènes pertinentes

Les signaux continus de type booléen peuvent être utilisés pour délimiter des segments avec début et fin correspondant à des situations remarquables. Sur ces segments sont calculés des attributs scalaires permettant de trier et/ou filtrer les instances de segments.

2.2.1 Cut-in sur VCS en section courante

- Description
 - Cet algorithme est déployé sur chaque instance de Cut-in détectée. On recherche en complément la présence d'une intersection à venir (voie d'insertion ou bretelle de sortie) grâce aux données cartographiques. En parallèle, est également testée la présence d'un virage à partir du signal continu calculé.
- Limites
 - La détection d'intersection s'appuie intégralement sur la base de données cartographiques et nécessite donc un positionnement correct de l'égo et une Base de données à jours.
- Sorties
 - Présence ou non d'au moins un virage pendant le Cut-in. (Bool)
 - Présence ou non d'au moins une intersection pendant le Cut-in (Bool)

2.2.2 Cut-in sur VCS en situation de suivi

- Description :
 - Algorithme permettant de vérifier la présence d'une *cible* devant l'égo avant le Cut-in. Dans le cas où une *cible* est effectivement détectée, l'algorithme a pour fonction de caractériser celle-ci.

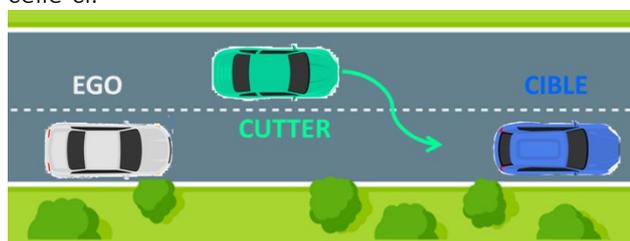


Figure 6 : Schéma d'un Cut-in avec Cible

La présence d'une *cible*, ainsi que son identifiant, est fournie en continu par la caméra intelligente. Sur VCS, la distribution des positions des *cibles* détectées est la suivante

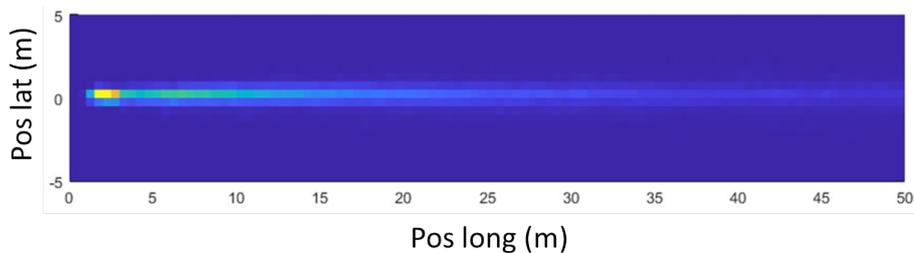


Figure 7 : Répartition des positions des cibles détectées sur VCS

On remarque que la quasi-intégralité des *cibles* sont détectées dans une bande centrée en 0 sur la position latérale ce qui est conforme à la définition de la *cible* (Objet dans la voie de l'Ego, devant l'Ego). Ainsi 98.8% des Cibles ont une position latérale comprise entre [-3m, 3m]. (Les exceptions interviennent lors de virages important). Par ailleurs 90% des cibles sont détectées à une position longitudinale de moins de 80m.

Note : A l'instant du Cut-in on a un changement de cible, celle-ci étant remplacée par le cutter qui devient donc la nouvelle cible.

- Sortie
 - Présence ou non d'une *cible* au moment du Cut-in (Bool)
 - Durée du suivie de la *cible* avant le Cut-in. Correspond au delta de temps entre l'instant où l'*égo* commence à suivre la *cible* et l'instant du Cut-in. (s)
 - Vitesse relative de la *cible* à l'instant précédent le Cut-in (m/s)
 - Position longitudinale de la *cible* à l'instant précédent le Cut-in (m). Correspond à la distance entre la *cible* et l'*égo*.
 - Temps Inter Véhicule (TIV) de la *cible* à l'instant précédent le Cut-in (s). Le TIV correspond au temps nécessaire à l'*égo* pour atteindre la même position que la cible. Il est calculé avec la relation suivante : $Tiv = \frac{Distance\ Cible}{Vitesse\ EGO}$

Note : Les résultats de cet algorithme sont utilisés à la fois pour sélectionner ET pour caractériser les scénarios sélectionnés.

2.2.3 Sections courantes pour l'analyse de l'impact de la présence de piéton

- Description :
 - L'algorithme utilise en entrée les segments temporels représentant la distance parcourue et l'orientation du véhicule ainsi que les instants durant lesquels la cartographie a indiqué la présence d'une intersection.
 - Il sélectionne les sections qui sont distants d'au moins 30 mètres d'une intersection et font une longueur maximale de 500 mètres.
 - Puis il supprime les sections qui ont une courbe trop importante, c'est-à-dire un changement de direction entre le début et la fin du segment supérieur de plus de 20 degrés.
- Limites
 - La détection d'intersection s'appuie intégralement sur la base de données cartographiques et nécessite donc un positionnement correct de l'*égo* et une Base de données à jours.
- Sorties
 - Liste de sections courantes

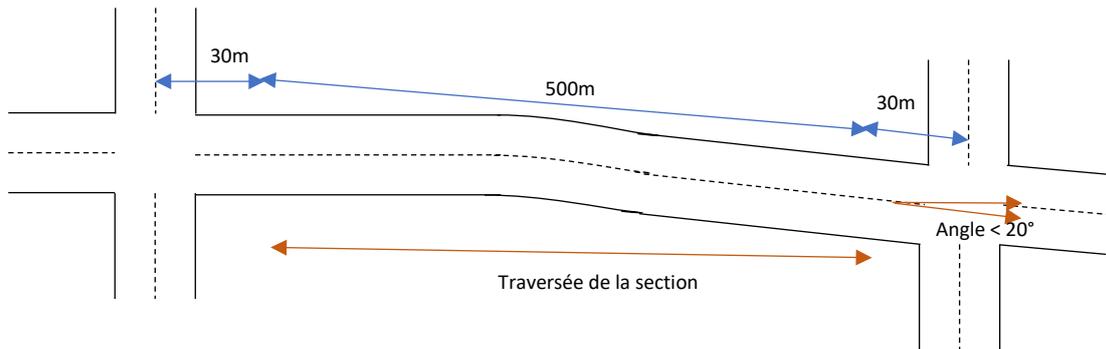


Figure 8 : Illustration des sections courantes

2.2.4 Intersections pour l'analyse de l'impact de la présence de piéton

- Description :
 - L'algorithme utilise en entrée les segments temporels représentant la distance parcourue ainsi que les instants durant lesquels la cartographie a indiqué la présence d'une intersection.
 - Il calcule des segments englobant les 30 mètres en amont et aval de l'intersection.
 - Si deux intersections sont distantes de moins de 30 mètres, elles sont concaténées.
- Limites
 - La détection d'intersection s'appuie intégralement sur la base de données cartographiques et nécessite donc un positionnement correct de l'égo et une Base de données à jours.
- Sorties
 - Liste des intersections

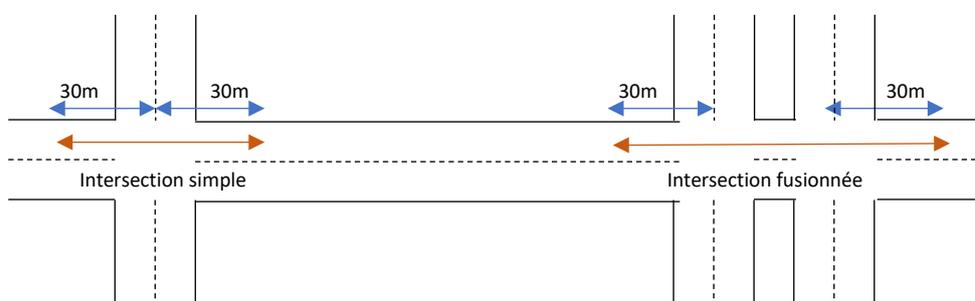


Figure 9 : Illustration des intersections

2.3 Caractérisation des scenarios

2.3.1 Cut-in sur VCS en suivi

Caractérisation du Cutter

- Description :
 - De façon analogue à la caractérisation de la *cible*, cet algorithme caractérise la trajectoire du *cutter*, à l'instant du Cut-in, C'est-à-dire au moment où le *cutter* devient la nouvelle *cible*
- Sorties :
 - Vitesse Relative du *cutter* à l'instant suivant le Cut-in (m/s)
 - Position relative du *cutter* à l'instant suivant le Cut-in (m). Correspond à la distance entre le *cutter* et l'*égo*.
 - Temps Inter Véhicule (TIV) du *cutter* à l'instant suivant le Cut-in (s).

Temps de rétablissement du TIV

- Description :
 - Lors d'un Cut-in, le *cutter* s'insère entre l'*égo* et la *cible*. La distance en mètre (où le TIV en seconde), est donc nécessairement réduite pendant le Cut-in. Ainsi, sur la base de données UDRIVE France, les Cut-in avec cible engendrent des diminutions de l'inter distance à la *cible* dont la distribution est représentée ci-dessous :

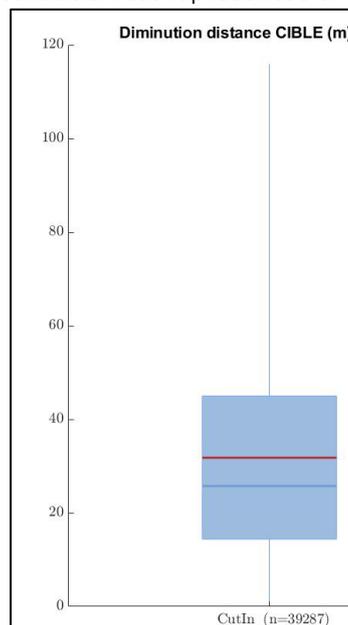


Figure 10 : Diminution des distances de sécurité pendant un Cut-in avec cible

En moyenne l'inter distance diminue de 31.7 m. Dans 75% des cas elle diminue de plus de 14.4 m.

A la suite de la manœuvre, l'*égo* peut chercher à rétablir sa distance de sécurité en adaptant sa vitesse. L'objectif de cet algorithme est de calculer la durée nécessaire à l'*égo* pour rétablir son inter distance.

- Sortie :
 - Durée de restauration du TIV (s). Correspond au delta de temps entre l'instant du Cut-in, et le premier instant où la distance entre la nouvelle *cible* (ancien *cutter*) et l'*égo*

est supérieure ou égale à la distance entre la *cible* et l'*égo* à l'instant précédent le Cut-in (calculée en 2.2.2). A noter que le TIV ne peut jamais être restauré si l'*égo* choisit de rester proche de la nouvelle *cible* ou si, par exemple, il subit un nouveau Cut-in réduisant encore plus son TIV. Dans ce cas la valeur de l'attribut sera celle par défaut, c'est-à-dire NaN (Not a Number), ce qui correspond à « valeur manquante ».

Début du Cut-in

- Description :
 - L'algorithme de détection du Cut-in permet de détecter le moment où le cutter devient la nouvelle *cible*. On définit l'instant du début de manœuvre comme étant l'instant où le *cutter* initialise son déplacement latéral menant au changement de voie et donc au Cut-in. La fonction de cet algorithme est donc de détecter le début du déplacement latéral du *cutter*, par la méthode suivante :
 - Au dernier instant où est détectée une vitesse latérale nulle de l'*obstacle* avant le rabattement ; c'est-à-dire au début du mouvement latéral du véhicule s'insérant.
 - Si une vitesse latérale nulle n'est jamais mesurée, alors le début de la manœuvre considéré est estimé au moment de l'apparition de l'*obstacle* par la caméra intelligente, dans la limite de 4s avant le franchissement.
 - Si l'*obstacle* est apparu plus de 4s avant le Cut-in, et que durant cette période aucune vitesse latérale nulle n'est détectée, alors le début de la manœuvre est estimée à 4s avant le Cut-in.

Ci-après divers exemples de détection du début de manœuvre, détecté par analyse de la trajectoire du *cutter*.

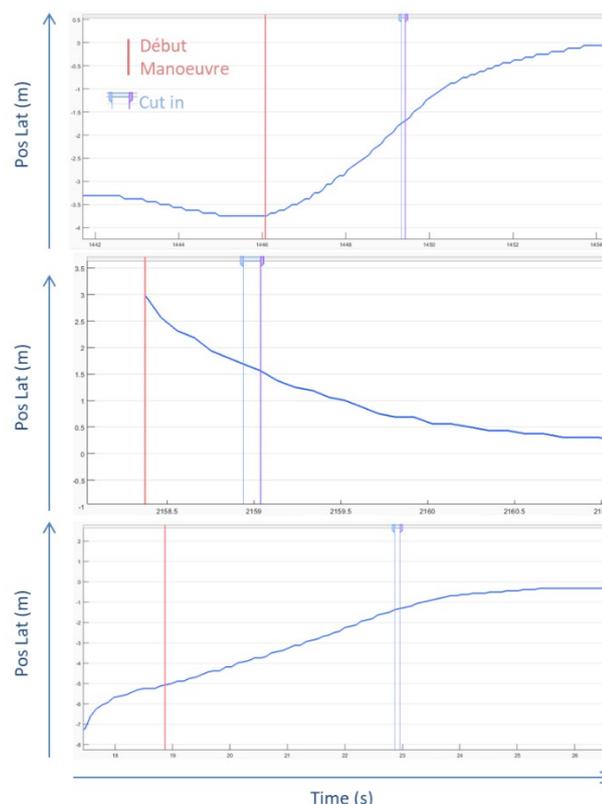


Figure 11 : Exemple de détection de débuts de manœuvre de Cut-in :

1. Détection d'une vitesse latérale nulle.
2. Apparition de l'obstacle cutter peu avant le Cut-in.
3. Début de manœuvre estimé 4s avant le Cut-in ;

- Sortie :
 - Début Manœuvre. Instant du début détecté de la manœuvre (Time)

Instants début de freinage

- Description :
 - Le comportement de l'égo pendant la manœuvre comprend une phase d'anticipation entre le début du Cut-in et le franchissement de ligne et une phase de réaction pendant les secondes suivant ce franchissement. La fonction de cet algorithme est de détecter le premier usage des freins durant ces 2 phases.
- Sorties :
 - Utilisation ou non des freins dans la phase anticipation (Bool)
 - Instant de première utilisation des Freins pendant la phase d'anticipation (Time)
 - Utilisation ou non des freins dans la phase de réaction (Bool)
 - Instant de première utilisation des Freins pendant la phase de réaction (Time)

Note : Les instant des freinages sont de type Time, signifiant qu'il s'agit d'instant dans l'enregistrement contenant la manœuvre de Cut-in. Ils ne peuvent être analysés qu'en les comparant à un autre instant se rapportant à cette manœuvre, typiquement l'instant du Cut-in, i.e. le changement de cibles.

Delta Vitesse Ego – Vitesse Maximale Autorisée (VMA)

- Description:
 - La fluidité du trafic sur VCS peut-être estimée par l'écart entre la vitesse du trafic et la VMA. En considérant que l'égo évolue dans le trafic et y adapte sa vitesse, on peut estimer sa vitesse égale à celle du trafic. La fonction de cet algorithme est de calculer la différence entre la vitesse de l'égo et la VMA qui est renseigné par la base de données cartographiques
- Limite :
 - La VMA renseignée par la base de données cartographiques peut-être erronée ou non à jour
- Sortie
 - Delta VMA : Différence entre la VMA et la vitesse de l'égo, à l'instant du Cut-in (km/h)

Temps à Collision Minimal

- Description :
 - Le temps à collision, désigné sous son acronyme anglais de TTC (Time To Collision), correspond à la durée restant avant une collision entre l'égo et un *obstacle*, si aucune mesure n'est entreprise pour l'éviter. Ces mesures peuvent être une adaptation de la vitesse de l'un ou des deux acteurs ou un évitement latéral. Le TTC est généralement calculé par rapport à la *cible*, c'est-à-dire à l'*obstacle* dans la voie de l'égo, juste devant celui-ci. Le TTC évolue en temps réel en fonction de la distance à la cible et de la vitesse relative de celle-ci :
$$TTC = \frac{\text{Distance Cible}}{\text{Vitesse relative CIBLE}}$$

Lors d'un Cut-in c'est le TTC par rapport au *cutter* qui est pertinent. Or, pendant la première partie de la manœuvre, celui-ci n'est pas encore la *cible*. La fonction de cet algorithme est de calculer le TTC entre l'égo et le *cutter*, y compris avant que celui-ci devienne *cible*, et de sélectionner la valeur minimum atteinte pendant toute la durée de la manœuvre.

- Sortie :
 - Temps à collision Minimale (s). Il s'agit du minimum atteint par le TTC pendant toute la durée du Cut-in :

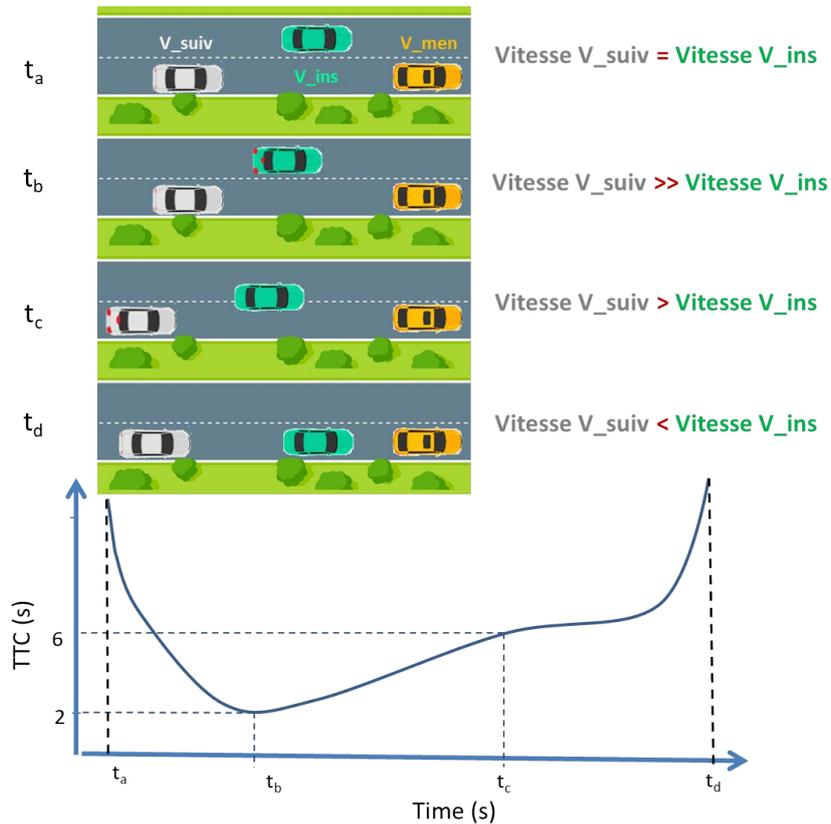


Figure 12 : Illustration du calcul de TTC entre l'égo et le cutter

Le découpage temporel de cette scène donne le scénario suivant :

- Lorsque $t = t_a$, le véhicule s'insérant et l'égo ne sont pas sur la même voie, le TTC ne s'applique pas. Il est cependant possible de calculer un TTC "virtuel" en considérant uniquement la distance et la vitesse relative longitudinale. Cette vitesse relative étant nulle (les 2 véhicules vont à la même vitesse) le TTC est infini.
- Lorsque $t = t_b$, suite à un freinage du véhicule s'insérant, la distance entre les 2 véhicules diminue et la vitesse relative est inférieure à 0, le véhicule suiveur va plus vite que le véhicule s'insérant. Le TTC peut donc prendre une valeur finie, en l'occurrence 2s sur l'illustration présentée. A noter qu'il s'agit du TTC "virtuel", les 2 véhicules n'étant toujours pas dans la même voie.
- Lorsque $t = t_c$, le véhicule s'insérant effectue sa manœuvre de rabattement, et rentre dans la voie de l'égo. Entre temps l'égo a adapté sa vitesse et le TTC est remonté à 6s.
- Lorsque $t = t_d$, le véhicule s'insérant à une vitesse supérieure ou égale à l'égo, leur TTC reprend une valeur infinie.

Durant tout cette situation, la valeur minimum du TTC a été atteinte avant que le véhicule s'insérant soit effectivement dans la même voie, lorsque $t = t_b$. C'est cette valeur de 2s qui sera retournée par l'algorithme.

Note : Cette explication est également présente dans le livrable 4.2 sur l'analyse des données.

Détection appels de Phare

- Description :
 - Les appels de phares, tels qu'ils ont été identifiés dans le signal continu correspondant peuvent marquer une volonté de l'égo de signaler sa présence. La fonction de cet algorithme est de détecter la présence ou l'absence d'appel de phase pendant la phase d'anticipation ou la phase de réaction constituant la manœuvre.
 -
- Sorties :
 - Appel de phare avant Cut-in. (Bool) Indique si les appels de phase ont été utilisés entre le début de la manœuvre et le franchissement de ligne du *cutter*.
 - Appel de phare après Cut-in. (Bool) Indique si les appels de phase ont été utilisés entre dans les 4s suivant le franchissement de ligne du *cutter*.

2.3.2 Caractérisation des Intersections et sections courantes pour l'analyse de l'impact de la présence de piéton

Pour les sections courantes les caractéristiques seront calculées sur toute la traversée de la section. Par contre pour les intersections nous aurons trois phases sur lesquelles les caractéristiques seront calculées : en traversée, en approche ou en sortie de l'intersection. Ces deux dernières phases sont situées 30 mètres avant le début de l'intersection et 30 mètres après la fin de l'intersection (voir Figure 13).

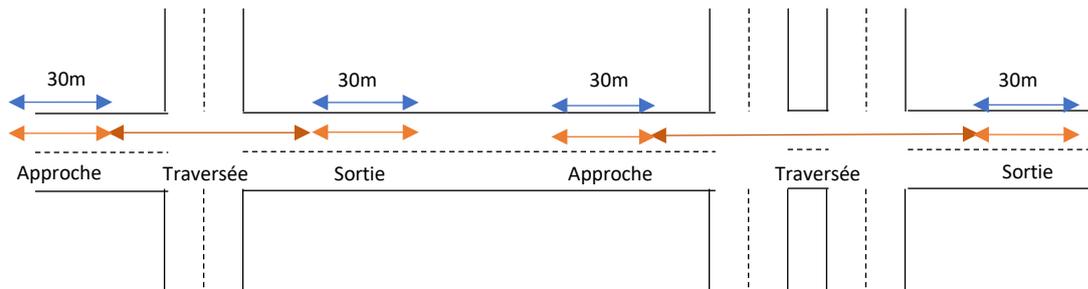


Figure 13 : Phases temporelles de calcul des caractéristiques sur les intersections

Caractéristiques de l'infrastructure

- Description : Ces caractéristiques permettent de définir deux attributs importants pour la gestion de la vitesse qui sont le type de route ainsi que la vitesse limite sur la section. Ces données proviennent du post-traitement de MapMatching qui fait correspondre à la position GPS du véhicule un segment de la base de données cartographiques dans laquelle des attributs sont définis pour chaque segment.
- Sorties communes aux intersections et sections courantes :
 - Type de route : cette caractéristique peut prendre les valeurs suivantes :
 - Urbain : correspond aux segments entre les panneaux d'entrée et de sortie d'agglomération.
 - Autoroute urbaine : voie à chaussée séparée en urbain.
 - Autoroute interurbaine : voie à chaussée séparée hors urbain.
 - Rampe : voie d'accès comme par exemple les bretelles d'entrée et de sortie d'autoroute ou entre deux nationales.
 - Rurale : autres sections hors urbain et voie d'accès ou autoroute.
 - Indéfini : lorsque le signal GPS n'est pas bon ou que le MapMatching n'a pas trouvé de segments dans la base de données cartographiques.
 - Vitesse maximale autorisée : décrit la VMA en km/h en début de section
- Sorties spécifiques aux intersections :
 - Nombre de voie dans la direction de l'ego en début d'intersection
 - Nombre de voie dans la direction de l'ego en fin d'intersection
 - Nombre de voie dans la direction opposée à l'ego en début d'intersection
 - Nombre de voie dans la direction opposée à l'ego en fin d'intersection
 - Giratoire (Bool) indique si l'intersection est ou n'est pas un giratoire
 - Longueur de l'intersection en mètre

Caractéristiques temporelles

- Description : Ces caractéristiques permettent de contextualiser les segments en termes de position dans le temps et de connaître la durée des sections.

- Sorties communes aux intersections et sections courantes :
 - Jour de la semaine : donne une des sept valeurs du jour de la semaine qui permettront de distinguer la semaine et le weekend
 - Heure Locale : permet de connaître l'heure du segment. Par exemple, combiner avec le jour de la semaine, cela permettra d'identifier les créneaux horaires durant lesquels il est probable que le trafic soit plus important (début et fin de journée dans la semaine), Nous définirons les créneaux suivants : semaine nuit (22H-6h), semaine début et fin de journée (7h-9h et 18h-19h), semaine journée (9h-17h), weekend nuit (22H-6h), weekend début et fin de journée (7h-9h et 18h-19h), weekend journée (9h-17h).
 - Période de la journée : cette caractéristique peut prendre les valeurs suivantes :
 - Journée : entre le lever du soleil (angle par rapport au zénith 90°) avec un azimut à l'est (90°) et le coucher du soleil (angle par rapport au zénith 90°) avec un azimut à l'ouest (270°).
 - Crépuscule : entre le coucher du soleil (angle par rapport au zénith 90°) et la fin du crépuscule nautique (angle par rapport au zénith $90 + 12 = 102^\circ$) avec un azimut à l'ouest (270°).
 - Nuit : entre la fin du crépuscule nautique (angle par rapport au zénith 102°) avec un azimut à l'ouest (270°) et le début de l'aube (angle par rapport au zénith 102°) avec un azimut à l'est (90°).
 - Aube : entre le début de l'aube (angle par rapport au zénith 102°) et le lever du soleil (angle par rapport au zénith 90°) avec un azimut à l'est (90°).
 - Temps du début de la section et temps de la fin de la section en seconde depuis le début d'enregistrement du trajet. Ces données permettront de calculer la durée de la section.

- Sorties spécifiques aux intersections :
 - Temps du début de l'approche de l'intersection en seconde depuis le début d'enregistrement du trajet.
 - Temps de fin de la sortie de l'intersection en seconde depuis le début d'enregistrement du trajet.

Caractéristiques de la dynamique du véhicule

- Description : Ces caractéristiques permettent de caractériser globalement le déplacement sur les segments en termes de vitesse.

- Sorties communes aux intersections et sections courantes :
 - Vitesse moyenne sur la phase de traversée de la section : cette donnée est la variable qui sera analysée pour estimer l'influence de la présence de piéton.
 - Pourcentage de temps pendant lequel le véhicule est arrêté (vitesse inférieure à 0.1 km/h). Ce paramètre permettra de sélectionner les sections sur lesquelles il n'y a pas d'arrêt.

- Sorties spécifiques aux intersections :

- Différence d'orientation de la position du véhicule entre l'entrée et la sortie de l'intersection en degré. Cette caractéristique permettra de savoir si le véhicule a fait un changement de direction dans cette intersection.
 - Vitesse moyenne en km/h dans la phase d'approche de l'intersection
 - Vitesse moyenne en km/h dans la phase de sortie de l'intersection
 - Accélération moyenne en m/s² dans la phase d'approche de l'intersection
 - Accélération moyenne en m/s² dans la phase de sortie de l'intersection
 - Accélération moyenne en m/s² dans la phase de traversée de l'intersection
- Sortie spécifique aux sections courantes :
 - Vitesse maximale sur la phase de traversée de la section

Caractéristiques liées aux autres usagers

- Description : Ces caractéristiques permettent de caractériser globalement les interactions possibles entre le véhicule et les autres usagers.
- Sorties communes aux intersections et sections courantes :
 - Pourcentage de temps pendant lequel le véhicule suit un autre véhicule dans la phase de traversée. Ce paramètre permettra de caractériser les sections avec suivi.
 - Pourcentage de temps pendant la traversée de l'intersection durant lequel il y a des piétons qui sont détectés sur ou au bord de la route. Ce paramètre permettra de caractériser les sections avec piétons.
- Sorties spécifiques aux intersections :
 - Pourcentage de temps pendant lequel le véhicule suit un autre véhicule dans la phase d'approche de l'intersection.
 - Pourcentage de temps pendant la phase d'approche de l'intersection durant lequel il y a des piétons qui sont détectés sur ou au bord de la route.
 - Pourcentage de temps pendant la phase de sortie de l'intersection durant lequel il y a des piétons qui sont détectés sur ou au bord de la route.

3 Base MOOVE

Pour rappel la base MOOVE est composée de données enregistrées à bord de 6 véhicules équipés de capteurs extéroceptifs capables de détecter et localiser les *obstacles* environnants :

- Une caméra intelligente (à l'avant),
- Un radar longue portée (à l'avant),
- 4 radars courte portée (sur les 4 ailes),
- 2 lidars Scala (à l'avant et à l'arrière).

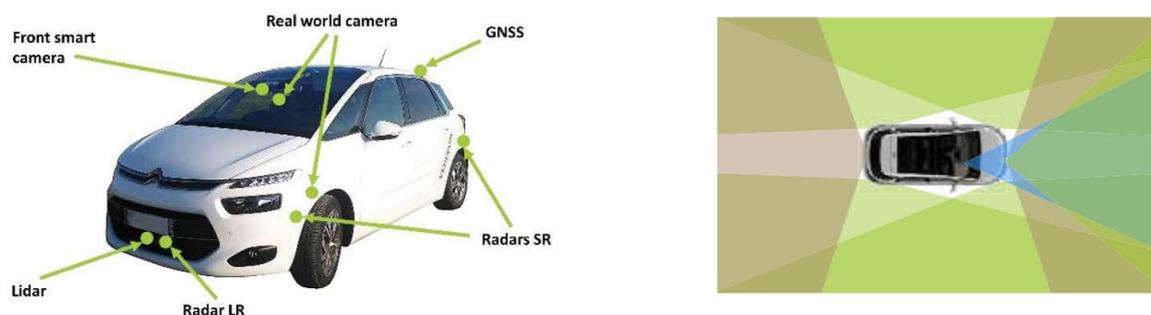


Figure 14 : Positions et angles de vue des capteurs extéroceptifs des véhicules MOOVE

En plus des capteurs, les données internes du Véhicule (CAN) et la géolocalisation de l'égo sont enregistrées. Une centrale inertielle complète cette instrumentation.

Les données de la base MOOVE ont été acquises sur route ouverte, en condition réelle de trafic alors que les véhicules étaient conduits par des chauffeurs professionnels. Les données enrichies pour les besoins du projet SURCA ont été collectées entre 2016 et 2018 principalement sur Voies à Chaussées Séparées, en région Parisienne.

De façon analogue aux données de la base UDRIVE l'enregistrement se fait en continu. L'enrichissement des données a également été effectué sur l'outil SALSA, et certains processus d'enrichissements mis au point sur la base UDRIVE ont pu être déployés sur la base MOOVE. C'est notamment le cas de la contextualisation de l'infrastructure routière par interrogation d'un Système d'Informations Géographiques (SIG).

3.1 Calculs de signaux continus

3.1.1 Fusion « Haut Niveau »

- Description :
 - La détection d'*obstacles* environnant l'égo se fait grâce à plusieurs capteurs percevant chacun une section de l'environnement. Afin de profiter de cette suite de capteurs, une opération complexe de « fusion haut-niveau » est nécessaire afin de mettre en commun toutes les perceptions. L'algorithme permettant cette fusion est composé de plusieurs étapes :
 1. **Recalage Spatial des capteurs.** Les capteurs détectent les *obstacles* dans leur propre référentiel. Afin de pouvoir fusionner les informations, il faut définir un repère commun à tous, propre à l'égo et effectuer les changements de repères correspondants. Le repère choisi a son origine au centre du parechoc avant du véhicule.
 2. **Filtre des détections.** Avant de fusionner les informations on supprime les détections éphémères et incohérentes qui durent moins de 0.5s. De plus seuls les *obstacles* identifiés comme mobiles sont fusionnés. Les obstacles

liés à l'infrastructure, constituant les bords de voie ou se situant au-delà sont ignorés.

3. **Identifications des détections communes.** Afin de déterminer que des détections de plusieurs capteurs se réfèrent à un seul et même *obstacle* on vérifie la cohérence spatiale et temporelle de celles-ci.
 4. **Calcul de l'*obstacle* résultant.** A partir des *n* détections originales se référant à 1 *obstacle*, on calcule à chaque timestamp la position de celui-ci. Les vitesses sont ensuite obtenues en dérivant les positions, les dimensions de l'*obstacle* sont déterminées puis figées sur toute la durée de vie de l'*obstacle*
 5. **Estimation de la trajectoire dans l'angle mort.** Lorsque l'*obstacle* passe de l'avant à l'arrière de l'*égo* (ou inversement), et que l'algorithme a fusionné les observations des différents capteurs correspondants, on peut alors estimer la trajectoire de l'*obstacle* lorsque celui-ci passe dans les angles morts de détection. Cette estimation de la trajectoire est particulièrement employée lors des remontées de file de 2RM.
- Limite :
 - La fusion ne permet pas d'exclure toutes les détections fausses/ imprécises des capteurs. De plus, la fusion par cohérence temporelle et spatiale des *obstacles* entraîne parfois des fusions « abusives » où plusieurs *obstacles* distincts sont fusionnés dans un seul.
 - Par ailleurs, si aucun capteur n'a vu un *obstacle* avec une confiance suffisante, celui-ci sera ignoré par l'algorithme de fusion et sera absent de la liste des *obstacles* détectés dans l'environnement de l'*égo*.
 - Sorties :
 - La sortie de l'algorithme de fusion contient toutes les données des *obstacles* de l'environnement de l'*égo*. Cela concerne tout particulièrement
 - Les positions dans le repère *égo*
 - Les vitesses, relatives et absolues
 - Le type d'*obstacle*
 - Le statut (En mouvement, à l'arrêt)
 - Ces signaux continus prennent à chaque instant autant de valeurs que le nombre d'*obstacles* détectés. Pour le besoin du projet SURCA, un sous ensemble des 12 obstacles les plus proches de l'*égo* est fourni.

3.1.2 Cut-in

- Description :
 - De façon analogue aux travaux menés sur la base UDRIVE, Cet algorithme a pour fonction de détecter l'insertion d'un véhicule (*cutter*) dans la voie de l'*égo*, devant l'*égo*. Un Cut-in est détecté si un *obstacle* passe par les étapes suivantes :
 1. L'*obstacle* roule sur le marquage délimitant la voie de l'*égo* (droite ou Gauche)
 2. Le centre de l'*obstacle* traverse le marquage et rentre sur la voie de l'*égo*)
 3. L'*obstacle* est identifié comme la nouvelle *cible* de l'*égo*
- Limite :
 - L'algorithme est sensible aux fausses détections d'*obstacles* résultant en des « *obstacles* fantômes » retourné par l'algorithme de fusion haut niveau mais

n'existant pas. Ces « *obstacles* fantômes » peuvent avoir des trajectoires transverses amenant à des faux positifs.

- L'algorithme de fusion comporte une correction de la courbure de la route afin de corriger le tracé de la route de la trajectoire naturelle de l'*obstacle*. En cas de virage important, et si l'*obstacle* se trouve à une distance suffisante, cette correction peut ne pas être suffisante et un Cut-in peut être détecté alors que l'*obstacle* est resté dans sa voie, en suivant le tracé de la route. Il convient cependant de préciser que seul des Cut-in éloignés de l'*égo*, et donc sans réels enjeux de sûreté, peuvent être faussement détectés par cette erreur

- Sorties :

- Timestamps de chaque Cut-in. Le début du Cut-in correspond au début du franchissement de ligne. La fin du Cut-in correspond au moment où le *cutter* finit le franchissement est qu'il est donc entièrement dans la voie de l'*égo* et devient la nouvelle *cible*.
- Présence ou non d'un Cut-in (bool)
- Id du *cutter*. Identifiant unique de l'*obstacle* réalisant la manœuvre

3.1.3 Remontée de file

- Description

- Cet algorithme a pour fonction de détecter à chaque instant les remontées de file subies par l'*égo* lorsqu'il évolue sur Voies à Chaussées Séparées. L'analyse de la trajectoire d'un *obstacle* permet de détecter une remontée de file si les conditions suivantes sont remplies :
 1. Pendant une phase à vitesse longitudinale relative non nulle, l'*obstacle* dépasse le véhicule. C'est-à-dire qu'il passe d'une position longitudinale inférieure à la longueur de l'*égo* (soit *position* = -4.2m) à la position au niveau du pare-choc avant du véhicule Ego (soit *position* = 0 m).
 2. Entre les « portes » virtuelles à -4.2 et 0m la position latérale de l'*obstacle* est comprise dans une bande de 1.5m de large à proximité droite ou gauche de l'*égo*.

- Limite

- La Fusion entre la détection arrière et avant des *obstacles* n'est pas toujours réalisée avec succès. Lorsque le lien n'a pas été établi, alors le véhicule est détecté en tant que 2 obstacles distincts et la remontée de file n'est pas identifiée.

- Sorties :

- Timestamps de chaque Remontée de file. L'instant de la remontée de file est le moment du passage de la « porte » virtuelle située au niveau du pare-choc avant de l'*égo*, centré sur le marquage droit ou gauche. Le signal obtenu est de type impulsion sa durée est réglé à 0.01s.
- Présence ou non d'une remontée de file
- Côté de la remontée de file (GAUCHE/DROITE)
- Id de l'*obstacle* effectuant la remontée de file.

3.2 Détections de scènes pertinentes

3.2.1 Type d'obstacles

- Description :
 - La distinction entre le type d'obstacle est un prérequis important du Projet SURCA, les interactions VA/2RM et VA/VL étant analysées dans des Work Package différents. La fonction de cet algorithme est d'identifier les classes des obstacles en fonction des résultats capteurs. Si le réel type d'un obstacle ne varie évidemment pas au cours du temps, son identification par un capteur peut en revanche évoluer. Par exemple, si un véhicule léger est obstrué par un autre obstacle, il peut être classifié en tant que 2 roues avant que cette classification ne soit corrigée lorsque l'angle de vue s'améliore. Pour déterminer la classe d'un obstacle, cet algorithme sélectionne le type qui a été le plus fréquent sur toute la durée de vie de l'obstacle.
- Limite :
 - La confiance dans le type d'obstacle identifié est d'autant plus grande que la détection est longue.
- Sortie :
 - Type de l'obstacle
 - Enumération (Voiture/ Deux roues/ Piéton/Camion/Bus)

3.3 Caractérisation des Scenarios

3.3.1 Scénario remontée de file

Le scénario remontée de file est caractérisé par des valeurs de signaux continus à certains moments clefs. Les signaux concernés sont :

- Vitesses relative et absolue
- Distance au marquage
- Positions relatives des *obstacles* par rapport à l'*égo*

Instants Clefs – Evènements

- Description
 - La fonction de cet algorithme est de détecter des évènements remarquables (également appelés instants clefs ou Key Times) intervenant durant le scénario. Les instants détectés par l'algorithme sont les suivants :

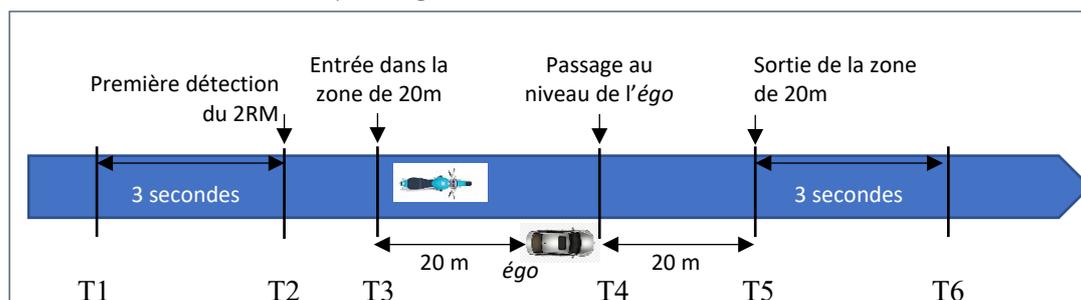


Figure 15 : Définition des instants T1 à T6 pour lesquels les données ont été extraites

- Sorties :
 - T2 : première détection du 2RM par les capteurs externes.
 - T3 : première détection du 2RM dans une zone de 20m autour du véhicule Ego. Il y a donc deux cas de figure : soit le 2RM est détecté avant qu'il ne rentre dans cette zone (à une distance supérieure à 20m vers l'arrière), dans ce cas T3 correspond à l'instant où le 2RM rentre dans la zone ($T3 > T2$), soit le 2RM est détecté alors qu'il est déjà dans la zone (à une distance inférieure à 20m), dans ce cas, T3 est l'instant de la première détection du 2RM ($T3 = T2$).
 - T4 : passage du 2RM au niveau de l'avant du véhicule Ego.
 - T5 : dernière détection du 2RM dans une zone de 20m autour du véhicule Ego (20 m devant le véhicule Ego).
 - Les instants T1 et T6 correspondent respectivement à 3 secondes avant T3, et 3 secondes après T5.

Note : Ces définitions sont également présentes dans le livrable 5.2 sur l'analyse des données des remontées de file de 2RM

3.3.2 Scénario Cut-in

De façon analogue, le scénario Cut-in est caractérisé par les valeurs de signaux continus à certains moments clefs. Les signaux concernés sont :

- Vitesses relative et absolue
- Positions relatives des *obstacles* par rapport à l'*égo*
- Temps Inter Véhicule
- TTC (Time To Collision)

Instants Clefs - Evènements

- Description
 - Les instants clefs du Cut-in sont définis de la façon suivante :

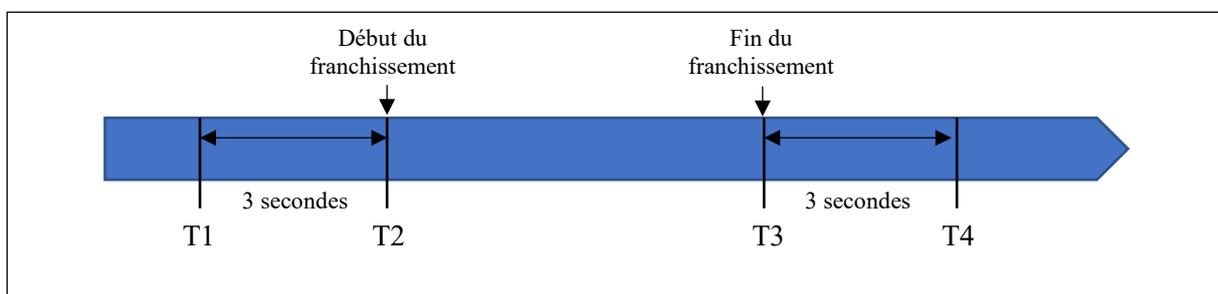


Figure 16 - Définition des instants T1, T2, T3 et T4

- T2 : instant de début du franchissement de la voie par la cible (le bord du véhicule empiète sur le marquage),
- T3 : fin du franchissement (le véhicule a entièrement franchi le marquage),
- Les instants T1 et T4 correspondent respectivement à 3 secondes avant T2, et 3 secondes après T3.

Décélération Moyenne

- Description :
 - L'impact du Cut-in sur la vitesse de l'égo permet de donner un indice sur la criticité de celui-ci et sur les mesures mis en œuvre par le conducteur pour gérer la situation. Afin de caractériser les Cut-in, en plus des valeurs instantanées de certains signaux, on calcule la perte de vitesse moyenne subi par l'égo durant la manœuvre avec la relation suivante :

$$\text{Décélération}_{EGO} = \frac{\text{Vitesse } EGO_{T3} - \text{Vitesse } EGO_{T2}}{T3 - T2}$$

- Sortie :
 - Décélération Moyenne de l'égo entre T2 et T3 (m/s²)

4 Base DYMOA

4.1 Rappel sur les données DYMOA

Le projet DYMOA est basé sur l'instrumentation d'une flotte d'une quarantaine de véhicules (30 2RM et 10 VL) à l'aide d'Enregistreurs de Données de la Route (EDR) de types smartphones. Le recueil, d'une durée de 1 an et demi, étendu sur 3 territoires (départements de Seine-Maritime, Bouches du Rhône et Hérault), concerne l'enregistrement des parcours des véhicules, d'incidents et d'événements détectés suite au dépassement de seuils de sollicitations dynamiques des véhicules, et de passages sur des zones d'intérêt prédéfinies (zones accidentogènes, aménagements spécifiques ...). Lors de ces situations, la dynamique du véhicule et les comportements des conducteurs sont enregistrés pendant 45s au moins, 30s avant le déclenchement d'événement/incident, ou l'entrée dans la zone d'intérêt, et 15s après. Les données enregistrées sont la vitesse et la trajectoire GPS à une fréquence de 1 Hz, les accélérations longitudinale, transversale et verticale et les vitesses de rotation de roulis, tangage et lacet à une fréquence de 100 Hz. Le contexte extérieur au véhicule est recueilli grâce à un enregistrement vidéo de la scène à l'avant pendant les 45s de la situation, avec une résolution volontairement dégradée afin d'avoir une bonne idée du contexte (interactions avec les autres usagers, configuration de la route, état de surface et visibilité) mais sans possibilité d'identifier les piétons et plaques minéralogiques. Des situations d'interaction VL/2RM correspondant aux scénarios critiques (L5.1) ont été sélectionnées à partir des vidéos d'incidents, d'événements et de zones d'intérêt des 2RM. L'instant de l'interaction est indépendant de l'instant de déclenchement dynamique ou géographique lié au projet DYMOA. La sélection de ces situations d'interaction correspond à l'étape de sélection de scènes pertinentes par algorithmes effectuées à partir des signaux continus sur les bases UDRIVE et MOOVE. L'ensemble des enrichissements effectués sur les données DYMOA concerne donc la caractérisation de ces scénarios identifiés.

4.2 . Caractérisation des Scenarios

La première étape consiste à reconstruire la situation d'interaction entre le 2RM dont on a les données réelles et le VL en interaction. L'évolution de ce dernier est visible sur la vidéo et il sera considéré par la suite comme *égo*. Une application a été développée sous Matlab pour simuler les évolutions simultanées des véhicules en interaction puis la détection du 2RM par l'*égo*. Les données utilisées pour le 2RM sont ses coordonnées et sa vitesse GPS, et son accélération longitudinale. Le programme permet de reconstruire la situation, puis de simuler une détection par le véhicule léger (VL). Les différentes étapes de reconstruction et de simulation sont les suivantes :

4.2.1 Reconstruction des trajectoires

- 1/ Conversion de la trajectoire GPS du 2RM, initialement en coordonnées WGS84 à 1 Hz, en coordonnées Lambert2E, et rééchantillonnage à une fréquence de 100 Hz.
- 2/ Lissage de la vitesse GPS également à 1 Hz et rééchantillonnage à 100 Hz.
- 3/ Affichage du plan des lieux issu de Google Street Maps et de la trajectoire du 2RM, et mise à l'échelle par itération jusqu'à obtenir une bonne adéquation (Figure 12). Les données du 2RM et du VL sont dans une feuille de classeur Excel (Figure 13) qui sont lues par le programme pour affichage en statique ou en dynamique. L'angle de lacet est calculé dans Excel d'après les coordonnées GPS du 2RM.

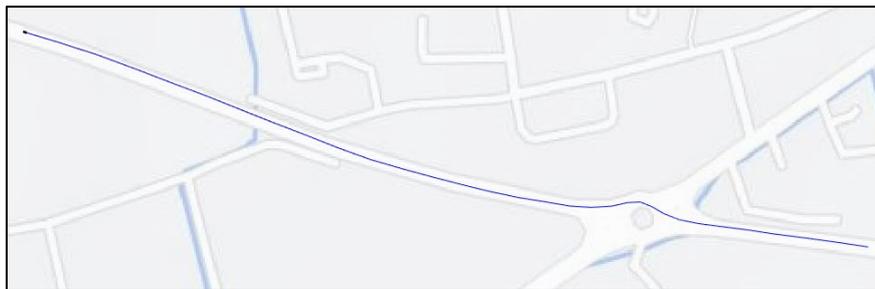


Figure 17 : Trajectoire du 2RM repositionnée sur une carte à l'échelle

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	t	Dist1	Speed1	Decel1	X1	Y1	Yaw1	Dist2	Speed2	Decel2	X2	Y2	Yaw2	length1	width1	length2	width2
2	-18.81	-145.1612688	0	0	52	16	28	-115.8597099	34.82	-0.14	185.34863	65.0662047	202.973719	4.64	1.8	2.1	0.85
3	-18.8	-145.1612688	0	0	52	16	28	-115.7584872	34.81	-0.18	185.258211	65.0277605	203.037251				
4	-18.79	-145.1612688	0	0	52	16	28	-115.6572644	34.81	-0.18	185.1679	64.989244	203.100876				
5	-18.78	-145.1612688	0	0	52	16	28	-115.5560417	34.81	-0.15	185.077698	64.9506554	203.164594				
6	-18.77	-145.1612688	0	0	52	16	28	-115.454819	34.8	-0.12	184.987605	64.9119947	203.228405				
7	-18.75	-145.1612688	0	0	52	16	28	-115.3535963	34.8	-0.11	184.897631	64.8733617	203.293265				

Figure 18 : Format des données dans Excel (temps, distance, vitesse, décélération, coordonnées GPS, angle de lacet, VL puis 2RM, longueur et largeur)

4/ Traçage de la position du VL au moment où les deux véhicules sont les plus proches, par itération. La vidéo et Google Earth sont utilisés par l'opérateur pour adapter au mieux la position relative des deux véhicules. Les deux véhicules sont représentés par des rectangles dont les longueurs et largeurs sont données par la feuille Excel (Figure 13).

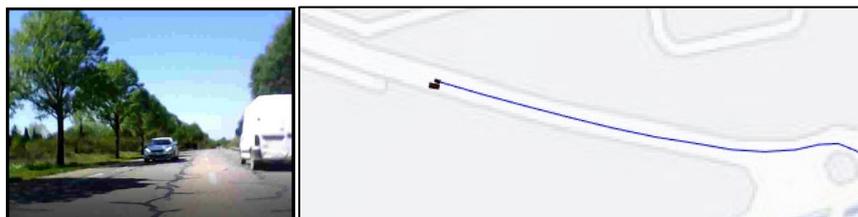


Figure 3 - Image vidéo avant le croisement - Position du VL antagoniste au moment du croisement

5/ L'évolution du VL avant et après la position déterminée au 4 est estimée par l'opérateur dans Excel, avec test d'hypothèses de vitesse, accélération, décélération, trajectoire. Le programme permet de simuler la situation en dynamique autant de fois que nécessaire, jusqu'à obtenir une simulation le plus proche possible de la situation enregistrée sur la vidéo (Figure 14).



Figure 19 : Trajectoires des deux véhicules en interaction

4.2.2 Simulation détection du 2RM

6/ Simulation de la situation d'interaction reconstituée, rejouée avec un cône de détection à l'avant du VL, et éventuellement un à l'arrière. Ces cônes de détection modélisent des caméras classiques. Une portée de 30 m et un champ de vue de 25° de part et d'autre de l'axe du VL ont été utilisés à titre illustratif (Figure 5, gauche). Le programme suit l'évolution des 4 coins et du centre du rectangle représentant le 2RM au fil du temps. Le 2RM est considéré comme détecté lorsque 3 points au moins parmi les 5 sont dans le cône de détection (Figure 15, droite).

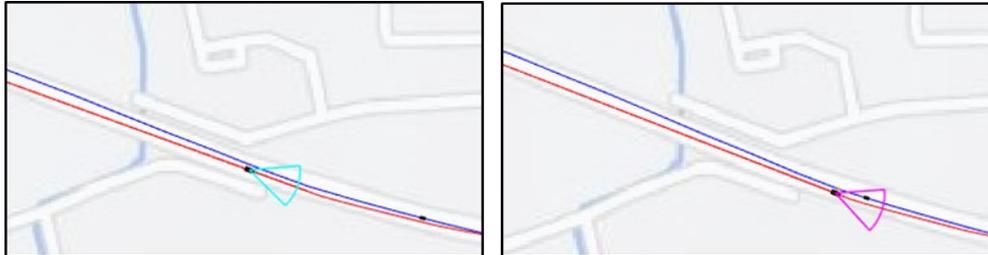


Figure 20 : Evolution avant détection et détection

7/ Les caractéristiques cinématiques des véhicules en approches sont ensuite relevées et leur inter distance calculée dans les derniers instants avant et pendant l'interaction.

5 Annexe

5.1 Glossaire des obstacles

- Le véhicule de collecte est nommé *égo*
- Les positions des *obstacles* sont exprimées dans un référentiel *égo*, sauf mention contraire.
- L'*obstacle* le plus proche devant l'*égo* et dans la voie de circulation de l'*égo* est appelé *cible*
- Pour les scénarios Cut-in le « *cutter* » et l'obstacle s'insérant devant l'*égo*

5.2 Type de données en sortie des enrichissements

- Les types de données obtenus en sortie d'algorithme sont les suivants
 - Temps. Exprimé en seconde il s'agit du nombre de seconde depuis le début de l'enregistrement. Il s'agit d'un nombre réel positif.
 - Enumération. Signal ou attribut prenant un nombre finit de valeurs (Exemple : Type de route)
 - Booléen. Signal ou attribut prenant la valeur 1 (VRAI) ou 0 (FAUX)