



ECOLE CENTRALE CASABLANCA
PROJET LEARNING BY DOING

SmartCross

CHIKHI Abdelilah
EL MRABET Aimane
FULGENCE ALPHONSE Badiane
DAHHASSI Chaymae
DOUMBIA Séao Andréas Diane

Encadré par : BOUHANI Hamza

20 juin 2023

Table des matières

0.1	Remerciements	1
0.2	Résumé	2
0.3	Liste des figures	3
0.4	Liste des tableaux	5
0.5	introduction	5
0.6	Développement du projet	5
0.6.1	Planification	5
0.6.2	Contexte et problématique	6
0.6.3	État de l'art	8
0.6.4	Évaluation des solutions existantes	11
0.6.5	Mise en place de la solution	12
0.6.6	Réalisation et tests	23
0.6.7	Résultats obtenus : Analyse et commentaire	36
0.7	Conclusion générale	36
0.8	Références bibliographiques	38

0.1 Remerciements

Nous souhaitons exprimer notre profonde gratitude envers notre tuteur, Monsieur BOUHANI Hamza pour son orientation et son encadrement précieux tout au long de ce projet. Grâce à ses conseils avisés, nous avons pu surmonter les défis et les problèmes rencontrés. Nous tenons à remercier également Madame Zerhouni Kaoutar pour son aide et son orientation dans la réalisation de la partie logicielle de notre projet.

Nous voulons également remercier chaleureusement les techniciens du Fablab, Monsieur DAGHOURI Abdelaziz et Madame ANNOUKOUBI Maha, dont l'expertise et l'assistance technique nous ont été d'une aide inestimable. Leur dévouement à nous soutenir dans la réalisation de ce projet a été exemplaire.

De plus, nous sommes extrêmement reconnaissants envers notre école pour nous avoir offert cette opportunité unique de concrétiser nos idées et de mettre en pratique nos connaissances, et aussi d'avoir mis à notre disposition le matériel nécessaire à la réalisation de notre projet.

Enfin, nous aimerions adresser nos sincères remerciements à Monsieur Khalid Dahi, le responsable du projet, pour sa vision et son dévouement qui nous ont aidé et orienté tout au long de ce parcours.

Nous sommes profondément reconnaissants envers toutes les personnes qui ont contribué à la réussite de ce projet, et nous les remercions du fond du cœur.

0.2 Résumé

Le projet Learning by Doing de cette année avait comme thème "Les villes intelligentes et durables en Afrique", et dans le cadre de ce thème intéressant, notre groupe a choisi de traiter la sous-thématique du transport et de mobilité intelligente, et plus précisément le problème de la congestion routière.

Les embouteillages dans les carrefours de la capitale sénégalaise, Dakar, posent un problème majeur, ils peuvent entraîner des retards importants pour les automobilistes et les usagers des transports en commun, ainsi qu'une augmentation de la pollution de l'air et des émissions de gaz à effets de serre.

Notre projet a donc pour objectif principal de résoudre ce problème d'embouteillage dans les croisements routiers Dakarois. Pour cela, nous avons entrepris le développement d'un système de feux de signalisation adaptatifs qui permettent de gérer intelligemment le trafic routier en ajustant la durée des feux en fonction du nombre de voitures présentes dans chaque voie.

Pour notre prototype, nous avons choisi d'utiliser, pour déterminer le nombre de voitures, une carte Raspberry Pi et des caméras , ainsi que le langage de programmation Python et les modules de reconnaissance appropriés tels que OpenCV et Roboflow. Cependant, étant donné que Dakar est une ville sujette à la poussière, nous avons également décidé d'intégrer des capteurs ultrason pour les périodes où la visibilité des caméras est réduite, tout en assurant la protection de nos caméras.

0.3 Liste des figures

Figure 1 : Diagramme de GANT résumant les différentes étapes du projet.

Figure 2 : Temps moyen perdu en trafic dans quelques villes de l’Afrique du Sud entre 2009 et 2016 d’après le TTTI.

Figure 3 : L’aménagement de la première ligne des ”voies lyonnaises” quai Claude Bernard a débuté et la majorité des écologistes de la ville et de Métropole de Lyon affirme qu’il n’y aura pas plus d’embouteillages.

Figure 4 : schéma représentant le réseau routier.

Figure 5 : schéma représentant l’ajout d’une route entre A et B.

Figure 6 : Solutions adoptées par l’État sénégalais.

Figure 7 : Diagramme bête à corne de notre système.

Figure 8 : Mécanisme générale de la solution adoptée.

Figure 9 : Détecteur de boucle.

Figure 10 : Caméra-vidéo installée sur un feu de signalisation.

Figure 11 : Capteur Radar.

Figure 12 : Capteur LIDAR.

Figure 13 : Capteur Ultrason.

Figure 14 : Caméra-vidéo utilisée dans le prototypage.

Figure 15 : capteur ultrason utilisé dans le prototypage

Figure 16 : Capteur de poussière GP2Y10 utilisé dans notre prototype.

Figure 17 : Module de feu de signalisation tricolore utilisé dans le prototype.

Figure 18 : Carte Raspberry pi 4 utilisée dans le prototype.

Figure 19 : Le servo-moteur utilisé dans notre prototype.

Figure 20 : Diagramme FAST du système.

Figure 21 : Les voitures de jeu utilisées dans notre prototype.

Figure 22 : l’outil ROBOFLOW utilisé pour la création du dataset.

Figure 23 : Exemple 1 d'une photo prise des voiture de jeu pour être utilisée comme élément du dataset.

Figure 24 : Exemple 2 d'une photo prise des voiture de jeu pour être utilisée comme élément du dataset.

Figure 25 : Exemple 1 de l'annotation d'une photo de voiture utilisée dans la dataset.

Figure 26 : Exemple 2 de l'annotation d'une photo d'un ensemble de voitures utilisée dans la dataset.

Figure 27 : les techniques de Data augmentation utilisée dans notre projet.

Figure 28 : Les résultats obtenus après la phase d'entraînement du modèle.

Figure 29 : Le code python qui permet d'utiliser le modèle entraîné sur de nouvelles photos.

Figure 30 : première photo utilisée pour tester le modèle entraîné.

Figure 31 : output du code de détection appliqué à la photo précédente où on voit clairement que le modèle était capable de détecter la voiture dans la photo.

Figure 32 : deuxième photo utilisée pour tester le modèle entraîné.

Figure 33 : output du code de détection appliqué à la photo précédente où on voit clairement que le modèle était capable de détecter les voitures dans la photo.

Figure 34 : La première partie du code de décompte appliquée à une photo de voiture de jeu.

Figure 35 : La deuxième partie du code de décompte appliquée à une photo de voiture de jeu, le résultat, affiché en bas, montre que le programme arrive à compter le nombre de voiture dans la photo.

Figure 36 : Un deuxième test du code sur une deuxième photo, on voit aussi que le programme arrive à compter le nombre de voitures dans la photo.

Figure 37 : Le code final

Figure 38 : Le plan des planches élaboré par le logiciel Laser cutting and engraving system

Figure 39 : Photo des deux planches de bois coupées par la découpe laser.

Figure 40 : Photo des tiges fixées sur le support principal du prototype.

Figure 41 : Assemblage des différentes composantes de la solution.

0.4 Liste des tableaux

Tableau 1 : Tableau comparatif des différentes solutions.

Tableau 2 : Fonction principale et fonctions contraintes.

Tableau 3 : Tableau comparant la caméra-vidéo, le radar et le LIDAR.

0.5 introduction

Dans le contexte de la mobilité urbaine croissante, les embouteillages routiers représentent un défi majeur pour de nombreuses villes à travers le monde. La ville de Dakar, capitale du Sénégal, ne fait pas exception, et les carrefours en particulier sont souvent congestionnés, entraînant des retards, des frustrations et une perte de productivité pour les citoyens.

De plus, les congestions routières ont un impact considérable sur l'environnement. Les véhicules immobilisés dans les embouteillages consomment davantage de carburant, contribuant ainsi à une plus grande émissions de gaz à effet de serre et de la pollution atmosphérique. Cette pollution de l'air a des conséquences néfastes sur la qualité de l'air que respirent les habitants, ainsi que sur les écosystèmes environnants.

C'est ainsi que nous avons essayé de trouver des solutions au problème de congestion, et plus particulièrement la congestion dans les carrefours routiers. Nous essayerons à travers ce rapport de présenter clairement le contexte et la problématique à traiter, un état de l'art qui présente les aspects théoriques nécessaires à la compréhension du problème et une description des solutions existantes, et enfin nous exposerons la solution retenue et les résultats obtenus.

0.6 Développement du projet

0.6.1 Planification

La planification du projet a été une étape essentielle pour assurer sa progression structurée et efficace. Avant même de commencer, nous avons consacré du temps à bien comprendre le thème de l'année "Les villes intelligentes et durables en Afrique", en examinant attentivement les différentes thématiques liées à ce sujet. Cette étape de contextualisation nous a permis de mieux comprendre le thème et cerner l'ensemble des problématiques qui en sont liées, ainsi que de connaître les différents enjeux et défis de chaque problématique.

Après avoir identifié l'ensemble des problématiques liées au thème de l'année, nous avons procédé à une étape cruciale qui est le choix de sujet. Après le choix d'une zone d'étude, à savoir la capitale Sénégalaise Dakar, nous avons mené des recherches pour identifier les multiples défis et problèmes auxquels cette ville est confrontée et leurs degrés de gravité. Cette étude nous a mené à traiter le problème de congestion routière, vue son importance et son impact considérable sur les citoyens de la ville et l'environnement.

La problématisation a suivi, au cours de laquelle nous avons étudié en profondeur les différents aspects et les implications du problème des embouteillages, plus particulièrement dans les carrefours de

Dakar. Cette étape nous a permis de mieux comprendre les facteurs qui contribuent à la congestion routière et de définir clairement les objectifs que notre projet devait atteindre.

Ensuite vient l'étape de la recherche de solution qui a été une phase intense où nous avons exploré diverses approches et technologies pour résoudre le problème des embouteillages. Nous avons évalué différentes options, en tenant compte des contraintes techniques, financières et environnementales.

Enfin, après le choix d'une solution à développer, nous avons procédé à la phase de prototypage, où nous avons mis en œuvre notre solution choisie pour tester sa faisabilité et son fonctionnement. Cette étape a été cruciale pour valider nos idées, ajuster les paramètres et affiner notre approche.

La planification rigoureuse de ces différentes étapes nous a permis de progresser de manière cohérente et méthodique, pour aboutir à la fin à la réalisation de notre prototype et la finalisation du projet.

Le diagramme de GANT suivant résume les différentes étapes du projet :



Figure 1 : Diagramme de GANT résumant les différentes étapes du projet

0.6.2 Contexte et problématique

Les embouteillages de circulation constituent un problème récurrent et préoccupant dans de nombreux pays africains, en particulier aux intersections (crossroads) des grandes villes. Ces congestions ont des conséquences néfastes sur la vie quotidienne des automobilistes et des usagers des transports en commun, engendrant d'importants retards et perturbations.

En effet, D'après le TomTom Traffic Index (TTTI) 2017 qui calcule le temps moyen perdu dans le trafic par jour, en comparant le temps de trajet supplémentaire pendant les heures de pointe par rapport à la situation de libre circulation, environ 42 minutes par jour sont perdues à cause du trafic au Cap, la capitale de l'Afrique du Sud, ce qui fait presque 163 heures par an. A Abidjan, une heure minimum par jour est perdue en cette même raison.[1]

La figure suivante donne le temps moyen perdu en trafic dans quelques villes de l'Afrique de Sud entre 2009 et 2016, d'après le TTTI :

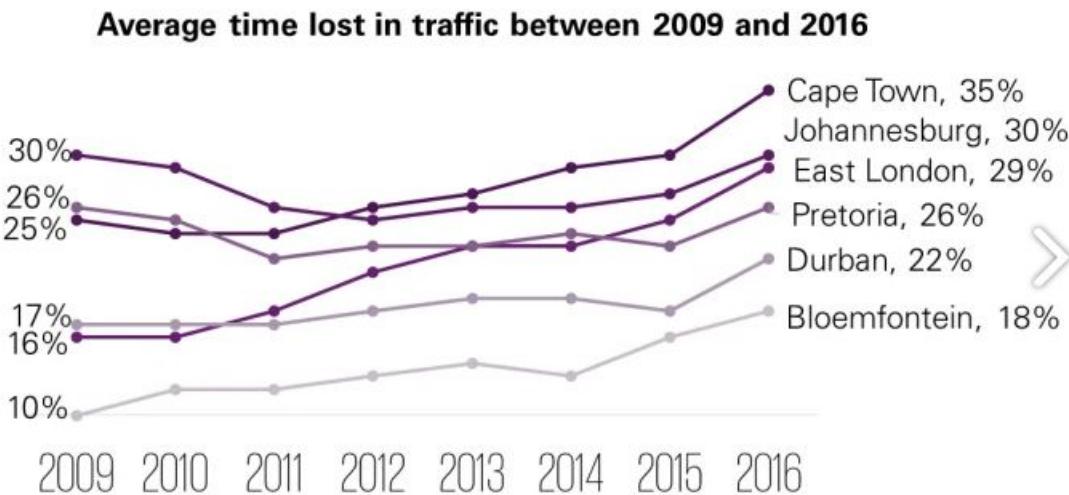


Figure 2 : Temps moyen perdu en trafic dans quelques villes de l'Afrique du Sud entre 2009 et 2016 d'après le TTTI

De plus, elles contribuent à une augmentation significative de la pollution de l'air et des émissions de gaz à effet de serre, impactant ainsi la qualité de l'environnement urbain et la santé des populations. En effet, entre les démarriages, les accélérations répétées, les freinages, la climatisation, etc, une voiture bloquée dans un ralentissement a tendance à consommer doublement le carburant. Par conséquent, la pollution et les émissions CO₂ doublent aussi. Ceci n'influe pas seulement sur l'environnement mais sur la santé publique aussi. Selon certains médecins interrogés sur le sujet, une épaisse fumée noire acre s'installe au fond des poumons, perturbant ainsi la respiration et pouvant générer des cas de cancer des voies respiratoires, un stress physiologique, une réduction de la fonction immunitaire,etc.[2]

En outre, les embouteillages sont un manque à gagner pour les transporteurs en Afrique. D'après **Romain Kouakou**, directeur général des Transports terrestres et de la circulation de Côte d'Ivoire, le coût lié aux « problèmes de circulation représentent près de 400 millions d'euros de perte par an », soit 5% du PIB, d'après La Banque Mondiale. C'est un constat qui se décline pratiquement dans chaque métropole africaine.[3][4]

Dans de nombreuses villes africaines, notamment dans les zones densément peuplées, les carrefours routiers sont des points de convergence où les flux de véhicules se croisent et se superposent, créant des goulets d'étranglement et des ralentissements. La situation est particulièrement critique dans des métropoles comme Dakar, la capitale sénégalaise, où la croissance démographique rapide, l'urbanisation intense et l'augmentation du parc automobile ont engendré une pression considérable sur les infrastructures de transport existantes.

Dakar, comme toute autre métropole africaine, n'échappe pas aux défis et problèmes causés par la congestion dans les croisements routiers. Face à ces défis, il devient impératif de trouver des solutions innovantes et durables pour gérer le trafic routier dans les intersections à Dakar. C'est dans ce contexte que notre projet se positionne, avec pour objectif de **développer un système de feux de signalisation adaptatifs qui permettra une gestion plus efficace et intelligente du trafic, en tenant compte des conditions réelles de circulation et en réduisant les temps d'attente aux carrefours.**

En combinant les avancées technologiques en matière de traitement d'image et de reconnaissance des véhicules, nous cherchons à améliorer la fluidité du trafic, à réduire les embouteillages et à pro-

mouvoir une mobilité urbaine plus efficace et durable à Dakar. Notre projet vise à répondre à la problématique des embouteillages aux intersections routières dakaroises, en contribuant à une meilleure qualité de vie pour les résidents, une économie plus dynamique et un environnement plus sain.

0.6.3 État de l'art

Dans cette partie, nous examinerons les solutions existantes pour la gestion des embouteillages aux intersections routières, en mettant l'accent sur les approches qui ont été adoptées dans des contextes similaires à celui de notre projet à Dakar, la capitale sénégalaise.

L'objectif est de comprendre les avancées réalisées jusqu'à présent dans ce domaine et d'identifier les forces et les faiblesses des solutions existantes. Cette revue de la littérature nous permettra de situer notre projet par rapport aux travaux antérieurs et de déterminer les opportunités d'innovation et d'amélioration. Nous examinerons notamment les différentes technologies utilisées, les méthodes de gestion du trafic, les systèmes de feux de signalisation adaptatifs et d'autres approches novatrices mises en œuvre dans des villes confrontées à des problèmes similaires. À travers cette analyse, nous serons en mesure de formuler une approche efficace pour notre projet de gestion des embouteillages à Dakar.

0.4.3.1 Système de feux de circulation adaptatifs

Contrairement aux feux traditionnels qui changent selon une minuterie, ces feux « intelligents » passent du rouge au vert et vice-versa selon le niveau de congestion, en temps réel.

D'après EnergyStream, un blog d'énergie des consultants Wavestone, cette solution a été mise en place à Bellevue (Washington D.C.), elle a permis de réduire le temps passé à rouler dans le quartier branché Factoria de Bellevue de 36% pendant les heures de pointe.[5]

Ce système fonctionne grâce à des capteurs de présence situés dans les routes qui permettent de détecter les voitures présentes pour ainsi permettre aux feux de s'adapter aux besoins de circulation.

En 2017, un tel projet a été lancé à Toronto. On a testé deux nouvelles technologies, la première, InSync, utilise des caméras pour évaluer le niveau de congestion, alors que l'autre, SCATS, a recours à des radars. Le coût par intersection est de 100000\$ à 150000\$.[6]

0.4.3.2 La tarification routière

La tarification de la route est une solution pour lutter contre les embouteillages en incitant les conducteurs à réduire leur utilisation de la voiture en augmentant le coût de l'utilisation des routes les plus encombrées.

Il existe plusieurs méthodes de tarification de la route, comme les péages urbains, les taxes sur les véhicules en fonction de leur empreinte carbone, le système de tarification kilométrique qui facture les conducteurs en fonction de la distance parcourue sur les routes, ou le système de tarification en fonction des heures de pointe qui facture les conducteurs qui utilisent certaines routes pendant les heures de pointe.[7]

Cette solution permet de réduire les coûts liés aux embouteillages comme les retards et les coûts de carburant, et peut également inciter les conducteurs à utiliser des modes de transport alternatifs tels que les transports en commun, les vélos ou la marche. En outre, les fonds provenant de la tarification routière peuvent être utilisés pour financer des projets de transport public ou des améliorations routières.

Cette taxation a été appliquée au centre-ville suédois depuis 2017. La région de 20% de la surface de Stockholm est entourée de postes de péage dont les tarifs augmentent en fonction des heures d'achalandage. Depuis sa mise en place, cette taxe anti-congestion a diminué le nombre de véhicules de 20 à 30%. [8]

0.4.3.3 Le transport en commun

En augmentant les options de transport en commun, comme les bus et les trains, les gens peuvent être incités à utiliser ces moyens de transport plutôt que de prendre leurs voitures, ce qui peut réduire le nombre de véhicules sur les routes aux intersections.

0.4.3.4 Les véhicules autonomes

La technologie s'est développée considérablement pour rendre les voitures plus autonomes. Plusieurs entreprises comme Google, BMW, Mercedes, Volvo et Tesla développent continuellement des systèmes permettant aux véhicules de rouler de manière autonome, c'est-à-dire sans l'intervention de l'humain. [9]

Des chercheurs français du CERMICS, Centre d'Enseignement et de Recherche en Mathématiques et Calcul Scientifique, misent sur la conduite autonome pour réduire les bouchons aux heures de pointe en harmonisant les flux de trafic.

Une centaine de véhicules autonomes ont sillonné les 6,4Km d'un tronçon autoroutier au sud de Nashville, au Tennessee aux États-Unis, pendant des heures où la circulation était la plus dense. Cette expérience menée en novembre 2022 et destinée à modifier le trafic pour réduire les bouchons et la consommation de carburant, résulte, en partie, du travail des chercheurs français. [10]

Les véhicules autonomes peuvent aider à réduire les embouteillages puisqu'il deviendra possible pour une voiture de prédire le comportement d'une autre, et ceci en utilisant des technologies de communication pour coordonner les déplacements des véhicules et en réduisant les erreurs humaines. Ainsi, ce fonctionnement autonome va permettre à la voiture d'éviter de trop ralentir à l'approche d'un feu si elle sait qu'il va passer dans quelques secondes au vert.

0.4.3.5 Infrastructures cyclables et piétonnières

En améliorant les infrastructures cyclables et piétonnières, les gens peuvent être incités à utiliser ces modes de transport plus écologiques et moins encombrants, ce qui peut réduire les embouteillages aux intersections.

La ville de Lyon a adopté ce modèle dans le cadre des aménagements des « voies lyonnaises », en remplaçant une voie de circulation sur le quai Claude-Bernard par une piste cyclable de 250Km. Une étude d'impact sur le trafic automobile et cycliste, menée par Fabien Bagnon, vice-président en charge des voiries et des mobilités au Grand Lyon, assure que cela « n'augmentera pas les embouteillages ». L'élu ajoute ainsi qu' « on ne peut pas supprimer la congestion automobile dans les grandes villes. Il faudra créer des infrastructures considérables que plus personne ne veut. La seule solution, c'est de favoriser des modes de déplacement plus efficaces ». [11]



Figure 3 :L'aménagement de la première ligne des "voies lyonnaises" quai Claude Bernard a débuté et la majorité des écologistes de la ville et de Métropole de Lyon affirme qu'il n'y aura pas plus d'embouteillages.

0.4.3.6 Construction de nouvelles routes

On craint parfois que la construction de nouvelles infrastructures routières ne fasse qu'augmenter la circulation. Mais elles peuvent quand même aider à désengorger les routes existantes et réduire les embouteillages aux carrefours.[12]

0.4.3.7 Le paradoxe de Braess

Le paradoxe de Braess cite que pour accélérer le trafic d'un réseau, il faut en retirer les portions les plus rapides.

En effet, les automobilistes adoptent un comportement égoïste, ils cherchent à prendre les voies les plus rapides et avec une densité de flux minimale. Par conséquent, une voie rapide déjà bouchée continuera de voir arriver des automobilistes, ce qui ne fera qu'empirer la situation.

Pour mieux comprendre, Suppossez que vous cherchiez à vous déplacer d'une ville (START) à une autre (END) plus vite possible. Deux itinéraires s'offrent à vous, l'un passant par A, l'autre passant par B. Le temps de parcours sur les nationales est de 45 minutes quel que soit la circulation. Cependant, comme il y a des feux sur les départementales, le temps de parcours dépend du nombre de voitures **n** qui l'emprunte : il vaut par exemple $\frac{n}{100}$ minutes.[13]

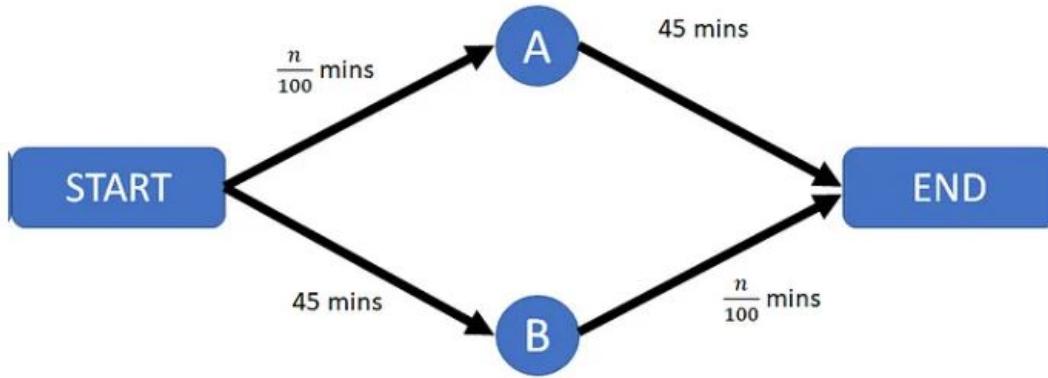


Figure 4 : schéma représentant le réseau routier

Supposons qu'il y ait 4000 voitures souhaitant passer de START à END. Si elles prennent toutes le trajet du haut (START-A-END) elles mettront 85 minutes. Idem si elles prennent toutes le trajet du bas. La situation optimale sera celle où le trafic s'équilibre entre les deux trajets. Le temps de parcours sera alors de $45 + \frac{2000}{100} = 65$ minutes.

Pour désengorger la circulation, on construit une voie ultra-rapide reliant A et B, dont le temps de parcours est négligeable quel que soit le nombre de véhicules qui l'emprunte (pas très réaliste mais bon c'est un modèle!). Avec cette voie rapide, le temps de parcours entre START et A devrait raccourcir, or c'est l'inverse qui se produit :

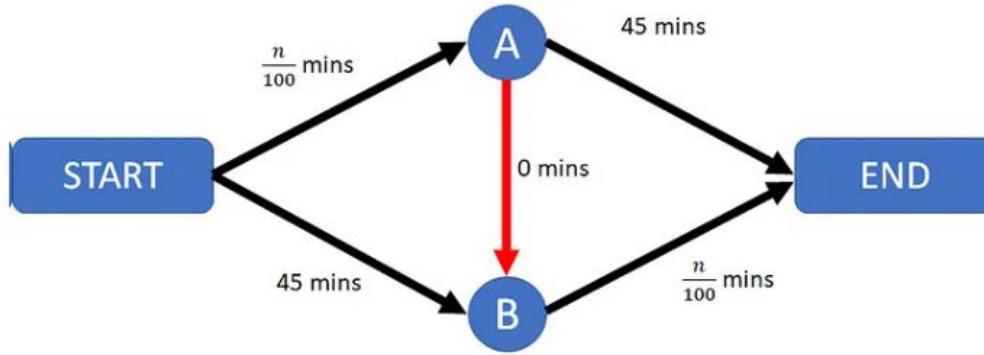


Figure 5 : schéma représentant l'ajout d'une route entre A et B

0.6.4 Évaluation des solutions existantes

L'évaluation des solutions existantes pour lutter contre les embouteillages est essentielle pour identifier les méthodes les plus efficaces et les plus adaptées aux besoins spécifiques et aux contraintes des villes africaines subsahariennes.

Nous procéderons ainsi à une catégorisation des solutions existantes selon les contraintes suivantes : **Contraintes organisationnelles, contraintes économiques et contraintes technologiques.**

0.4.4.1 Contraintes organisationnelles : le transport public

De nombreuses entreprises de transport public appartenant et gérées par l'État en Afrique subsaharienne ont cessé d'exister au début des années 1990 en raison des politiques d'ajustement structurel qui ont imposé aux pays de l'Afrique subsaharienne de se conformer aux programmes d'aide associés aux agences internationales, en particulier au Fonds Monétaire International (FMI).[14]

Aujourd'hui, dans toute l'Afrique subsaharienne, le transport public est en grande partie dominé par les opérations du secteur informel désorganisé. La dominance de ces services désorganisés entrave le développement économique et réduit la qualité de vie des citoyens car le grand nombre de véhicules nécessaires pour répondre à la demande cause non seulement des embouteillages, mais également des problèmes de stationnement, de pollution et de faibles niveaux de sécurité routière.[14]

0.4.4.2 Contraintes économiques

Les solutions telles que la construction de nouvelles routes et des infrastructures cyclables et piétonnières peuvent comporter des limitations financières importantes pour les pays africains.

La construction de nouvelles routes coûte cher en particulier dans les zones rurales ou densément peuplées où les terrains peuvent être difficiles à acquérir ou les infrastructures déjà présentes peuvent poser des défis.[14]

En effet, d'après l'Administration fédérale des routes (FHWA) des États-Unis, le coût moyen par mile pour construire une nouvelle route rurale à deux voies est compris entre 3 et 30 millions de dollars, le coût pour construire une nouvelle autoroute urbaine peut être encore plus élevé, avec un coût moyen compris entre 15 et 150 millions de dollars. Dans les pays en développement comme les pays africains, le coût pour construire une nouvelle route peut être moins à celui des pays développés, mais il peut encore être très élevé surtout lorsqu'on tient compte des coûts d'acquisition des terrains et de la main d'œuvre.[14]

0.4.4.3 Contraintes technologiques et économiques

Adopter des solutions comme la tarification des routes et les voitures autonomes s'avère à la fois économiquement coûteux, et difficile à implémenter par manque des moyens technologiques adéquats.

En effet, il y a un manque de capacité technique pour mettre en place des solutions de transport intelligentes comme les systèmes de gestion de la circulation, les systèmes de transport en commun automatisés et les véhicules autonomes. Cela est souvent dû à un manque de personnel qualifié et de formation adéquate dans ces domaines.

Selon un rapport de Manpower Group[15], l'Afrique du Sud manque de personnel qualifié pour remplir les rôles essentiels associés à la construction d'une économie forte.

0.6.5 Mise en place de la solution

0.4.5.1 Solutions mises en oeuvre par l'État sénégalais

Une étude réalisée en 2021 par le média Makers Africa[16], classe Dakar dans le top 5 des villes africaines les plus embouteillées. Les bouchons à Dakar constituent un vrai cauchemar. Les routes de la capitale sénégalaise sont asphyxiées en matinée comme en soirée, et surtout aux heures de pointe.

Impossible de faire un trajet sans subir le spectre des ralentissements quotidiens.

Dakar possède un réseau de transport aussi bien public que privé. Entre les bus Dakar Dem Dikk, les Ndiaga Ndiaye, les cars rapides, les taxis moto, les taxis et le TER, récemment inauguré, etc. Dakar est assez bien équipée pourtant elle est embouteillée. Néanmoins, pour mettre fin à ces bouchons monstrueux qui coûtent par ailleurs plus de 100 milliards de FCFA par an à la ville de Dakar, l'État a mis en place plusieurs solutions : le **TER (train express régional)**, le **BRT (Bus Rapid Transit)**, projet lancé depuis 2019 et qui a comme objectif réduire de plus de la moitié la durée du transport à Dakar, et enfin des pistes cyclables ont été mises en place pour favoriser la pratique du vélo au Sénégal.[16]

On peut donc résumer les solutions mises en oeuvre par l'État sénégalais comme suit :



Figure 6 : Solutions adoptées par l'État sénégalais

0.4.5.2 Solution choisie par notre groupe de projet

Malgré les mesures adoptées pour lutter contre la congestion routière à Dakar, et les milliards de Francs CFA dépensés pour la mise en place de solutions comme le TER et le BRT (120 milliards de FCFA pour le TER [17] et 300 milliards de FCFA pour le BRT [18]) , le trafic dakarois est encore à plaindre et fait perdre 108 milliards FCFA à l'État. [19]

Afin de contribuer à l'apaisement du problème de la congestion routière dans les croisements à Dakar, et pour prendre en considération les contraintes de temps, de budget et de connaissances qui limitent la réalisation de notre projet, nous avons recouru à un tableau comparatif pour comparer les différentes solutions possibles.

Solutions	Faisabilité	Durée	coût	Technicité
Tarification de la route	Faisable	Moyenne	Moyen	Moyenne
Paradoxe de Braess	Faisable	Longue	Elevé	Faible
Véhicules autonomes	Complexe	Longue	Elevé	Très élevée
Feux de signalisation adaptatifs	Faisable	Moyenne	Moyen	Moyenne
Construction des routes	Faisable	Longue	Elevé	Elevée
Infrastructures piétonnières	Faisable	Longue	Elevé	Elevée
Transport en commun	Complexe	Longue	Elevé	Elevée

Tableau 1 : Tableau comparatif des différentes solutions

Ainsi, en tenant compte de ce qui précède, et dans le but d'avoir des résultats plus satisfaisants, on a opté pour **l'utilisation d'un système de feux de circulation adaptatifs** qui font circuler les véhicules plus rapidement et avec moins d'arrêts. Ces signaux sont constamment programmés pour maximiser la durée du feu vert et permettre le passage du plus grand nombre de voitures au lieu d'adhérer à un cycle fixe, comme la majorité des systèmes de circulation utilisés.

Cependant, à travers notre recherche sur le terrain dans la ville de Dakar, nous avons constaté que celle-ci est sujette à des tempêtes de poussière, en particulier pendant la période estivale. Cette observation a été essentielle pour adapter notre solution aux conditions spécifiques de la ville et ainsi proposer une solution adaptée aux conditions climatiques de la ville de Dakar.

En tenant compte de ce facteur, nous avons développé des stratégies visant à garantir le bon fonctionnement de notre système de gestion des embouteillages, même dans des conditions de visibilité réduite causées par la présence de poussière. En intégrant des capteurs ultrason supplémentaires et en prenant des mesures de protection pour nos caméras, nous avons assuré la fiabilité et la performance continues de notre solution, même dans des conditions environnementales difficiles. Cette adaptation aux caractéristiques propres à Dakar renforce l'efficacité de notre projet et sa capacité à faire face aux défis spécifiques rencontrés dans cette ville dynamique.

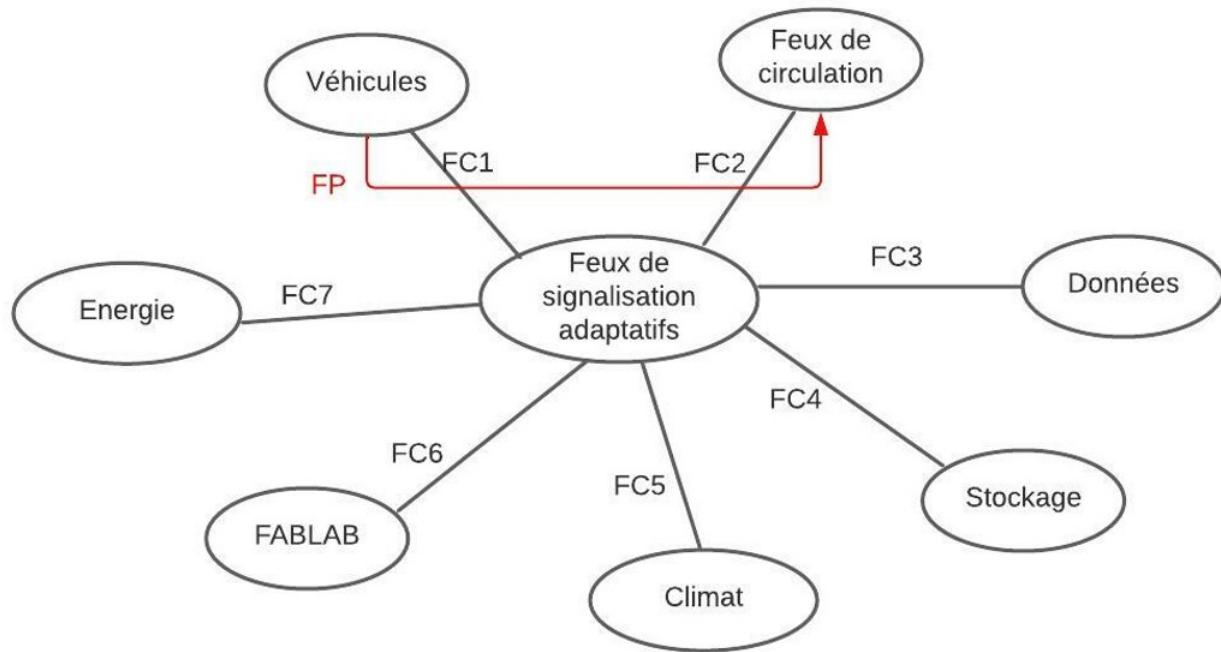


Figure 7 : Diagramme bête à corne de notre système

FP	Doit être capable de changer la durée des feux tricolores en fonction de la densité du trafic routier
FC1	Doit être capable de détecter les véhicules
FC2	Doit commander les feux de signalisation
FC3	Doit traiter les données reçues à l'aide des algorithmes
FC4	Doit être capable de stocker des données
FC5	Doit résister aux changements climatiques
FC6	Doit être conforme aux contraintes matérielles du FABLAB
FC7	S'alimenter avec une énergie électrique

Tableau 2 : Fonction principale et fonctions contraintes

0.4.5.3 Technologies utilisées pour l'implémentation de la solution

D'un point de vue global, notre solution repose sur le mécanisme suivant :



Figure 8 : Mécanisme générale de la solution adoptée

Dans cette partie du rapport, nous allons essayer de citer les résultats de nos recherches sur les technologies disponibles pour la réalisation du système de feux de signalisations adaptatifs, comparer entre les technologies trouvées pour enfin préciser le choix retenu par notre groupe.

0.4.5.3.1 Détection des véhicules

Afin de détecter la présence des véhicules dans le carrefour, il est envisageable d'utiliser des capteurs.

Il existe plusieurs types de capteurs qui peuvent être utilisés pour détecter les véhicules dans un système de feux de signalisation intelligent :

- **Détecteur de boucle** : Ces capteurs sont installés dans la chaussée et détectent les véhicules en utilisant des boucles électromagnétiques. Les détecteurs de boucle peuvent être utilisés pour détecter les véhicules à des fins de comptage, de classification et de surveillance de la vitesse.

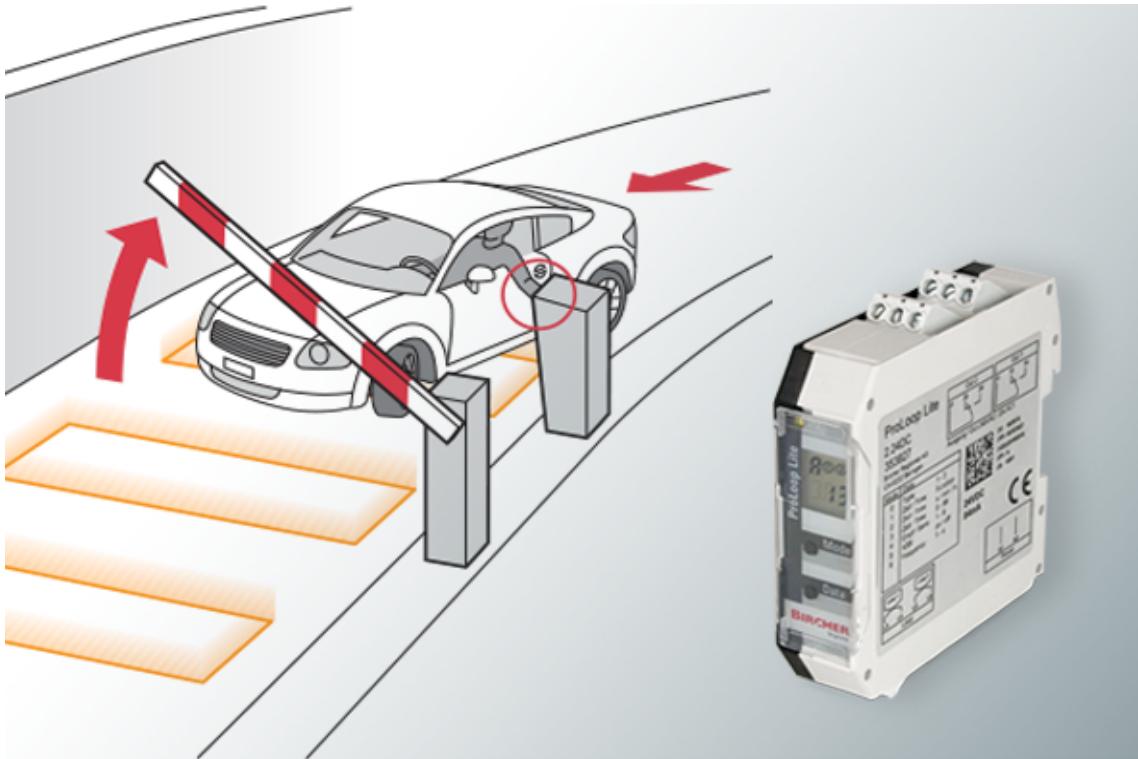


Figure 9 : DéTECTEUR DE BOUCLE

- **Caméra-vidéo :** Les caméras vidéo peuvent être utilisées pour détecter les véhicules en utilisant la reconnaissance d'image ou la reconnaissance de formes. Les caméras peuvent être utilisées pour compter les véhicules, identifier leurs types et suivre leurs trajectoires[20].



Figure 10 : Caméra-vidéo installée sur un feu de signalisation

- **Capteurs Radar** : Les capteurs radar peuvent être utilisés pour détecter les véhicules en utilisant des ondes radio.

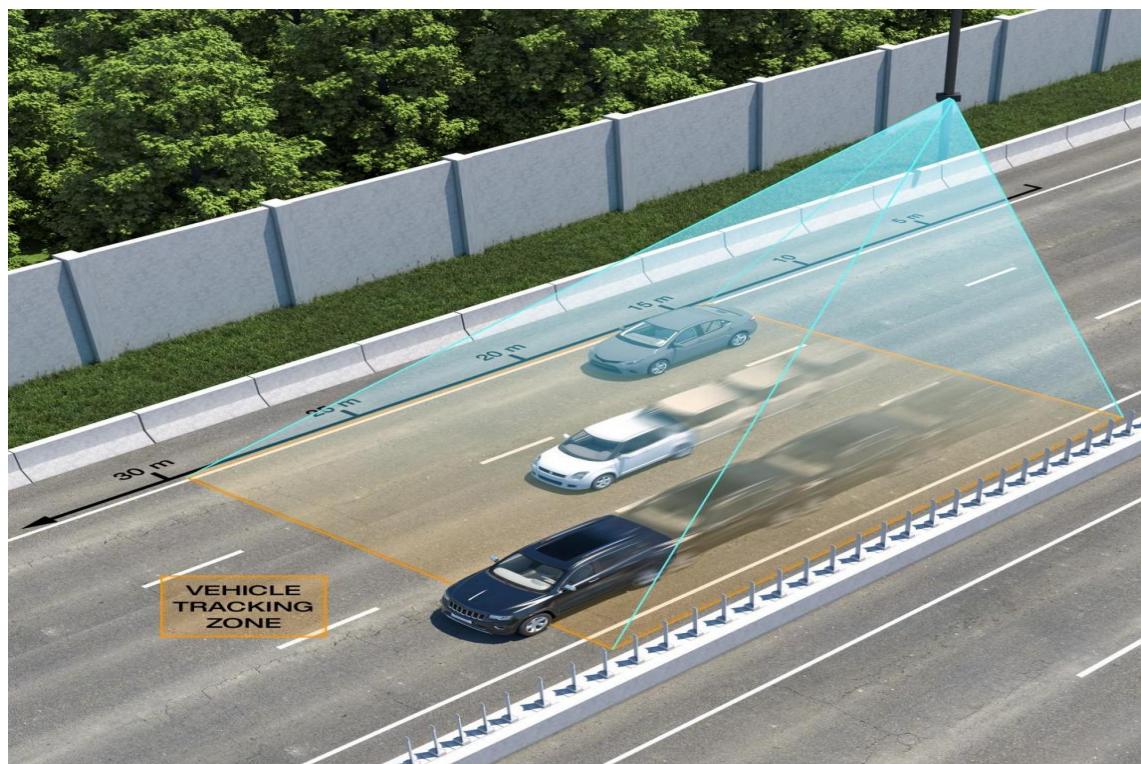


Figure 11 : Capteur Radar

- **Capteurs LIDAR** : Les capteurs LIDAR (Light Detection And Ranging) utilisent un laser pour mesurer la distance et la vitesse des objets. Il peut être utilisé pour détecter les véhicules, les cyclistes et les piétons.



Figure 12 : Capteur LIDAR

- **Capteur Ultrason** : Les capteurs Ultrason peuvent être utilisés pour détecter les véhicules en utilisant des ondes sonores.



Figure 13 : Capteur Ultrason

Afin de choisir la meilleure solution technologique qui permet de détecter les véhicules et qui répond aux exigences budgétaires, nous avons recouru à un tableau comparatif où on a comparé entre la caméra-vidéo, le radar et le LIDAR.

Capteurs	Avantages	Inconvénients
Capteur vidéo (caméra intelligente)	<ul style="list-style-type: none"> Distingue facilement les formes et les couleurs et identifie rapidement le type d'objet sur la base de ces informations. Faible coût. 	<ul style="list-style-type: none"> Mauvaise vision lors des événements météorologiques extrêmes.
Radar	<ul style="list-style-type: none"> Insensitive aux conditions météorologiques. 	<ul style="list-style-type: none"> N'identifie pas exactement l'objet détecté.
LiDAR	<ul style="list-style-type: none"> Modélisation 3D haute définition. 	<ul style="list-style-type: none"> Nécessite une puissance de calcul importante. Coût élevé. Mauvaise performance dans les événements météorologiques extrêmes (pluie...)

Tableau 3 : Tableau comparant la caméra-vidéo, le radar et le LiDAR

Ainsi, d'après ce qui précède, on a opté au début pour l'utilisation des caméras-vidéo pour la détection des voitures pendant les conditions météorologiques normales, puis basculer vers l'utilisation d'un radar

pendant les tempêtes de poussière. Néanmoins, il s'est avéré après, pendant la phase de prototypage, que les radars sont des dispositifs puissants qui permettent de détecter les objets sur une zone d'espace très large (20 mètres), offrant ainsi une capacité étendue de détection des objets dans un large champ de vision, chose qui nous posera problème vue les petites dimensions de notre prototype.

C'est ainsi qu'on a décidé d'utiliser **des capteurs ultrason** à la place des radars, ces capteurs qui ont aussi l'avantage d'être insensibles aux conditions climatiques mais qui sont moins puissants et qui seront donc parfaitement adaptés à notre prototype.

Solution retenue :

En tenant compte de ce qui précède, et vue les conditions climatiques spéciaux de la ville de Dakar, nous avons retenu la solution suivante : **Utiliser une caméra-vidéo pour détecter les voitures quand les conditions climatiques sont normales, puis basculer vers l'utilisation d'un capteur ultrason, qui a l'avantage d'être insensible aux conditions météorologiques, quand le climat devient poussiéreux.**



Figure 14 : Caméra-vidéo utilisée dans le prototypage



Figure 15 : capteur ultrason utilisé dans le prototypage

0.4.5.3.2 Détection de la poussière

Pour surveiller efficacement la présence de poussière lors des tempêtes fréquentes à Dakar, nous avons intégré un capteur spécifique de poussière dans notre système (GP2Y10).

Ce capteur de poussière est conçu pour détecter et mesurer la concentration de particules en suspension dans l'air et donc informer en temps réel sur le niveau de poussière environnant. Cela nous permet de prendre des mesures appropriées pour adapter notre solution aux conditions de visibilité réduite

causées par les tempêtes de poussière en basculant vers l'utilisation du capteur ultrason et en activant la protection des caméras-vidéo.

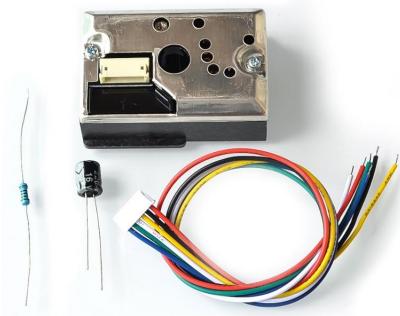


Figure 16 : Capteur de poussière GP2Y10 utilisé dans notre prototype

0.4.5.3.3 Les feux de signalisation

Dans notre prototype, nous avons utilisé des modules de feu de circulation tricolore.



Figure 17 : Module de feu de signalisation tricolore utilisé dans le prototype

0.4.5.3.4 Contrôleur de signalisation

Les contrôleurs de signalisation sont des dispositifs électroniques qui reçoivent les données des capteurs et déterminent la durée des feux verts et rouges en fonction de la quantité de circulation. Ces contrôleurs peuvent être installés soit sur les feux de signalisation eux-mêmes, soit dans un boîtier séparé.

Dans notre cas, en nous appuyant sur les ateliers de prototypage rapide qui figurent dans notre programme de formation, on a opté pour l'utilisation des contrôleurs de signalisation basés sur des systèmes embarqués et plus précisément les micro-ordinateur. On a donc utilisé dans notre prototype une carte Raspberry Pi qui permettra, en s'appuyant sur un programme convenable, analyser les données des capteurs et commander les feux de signalisation en conséquence.



Figure 18 : Carte Raspberry pi 4 utilisée dans le prototype

0.4.5.3.5 Protection des caméras

Comme mentionné auparavant, notre solution sera adaptée aux conditions climatiques spéciaux de la ville de Dakar, et donc, pour éviter l'endommagement des caméra lors des tempêtes de poussière, nous avons envisagé de mettre en oeuvre un système de protection qui utilisera principalement des **servo-moteur SG90** qui permettront de déployer un voile de protection sur la caméra.



Figure 19 : Le servo-moteur utilisé dans notre prototype

0.4.5.3.6 La partie logicielle de la solution (Software)

Afin d'analyser les données recueillies des différents capteurs et commander les différents composants de la solution, nous avons recouru au langage de programmation **Python** qui est un langage performant pour le traitement des images et la reconnaissance des objets.

Les différentes librairies et méthodes du machine-learning utilisées pour la mise en oeuvre de la solution seront traitées dans la partie suivante "Réalisation et tests".

0.4.5.3.7 Résumé

Le diagramme FAST suivant permet de résumer les différentes composantes du système :

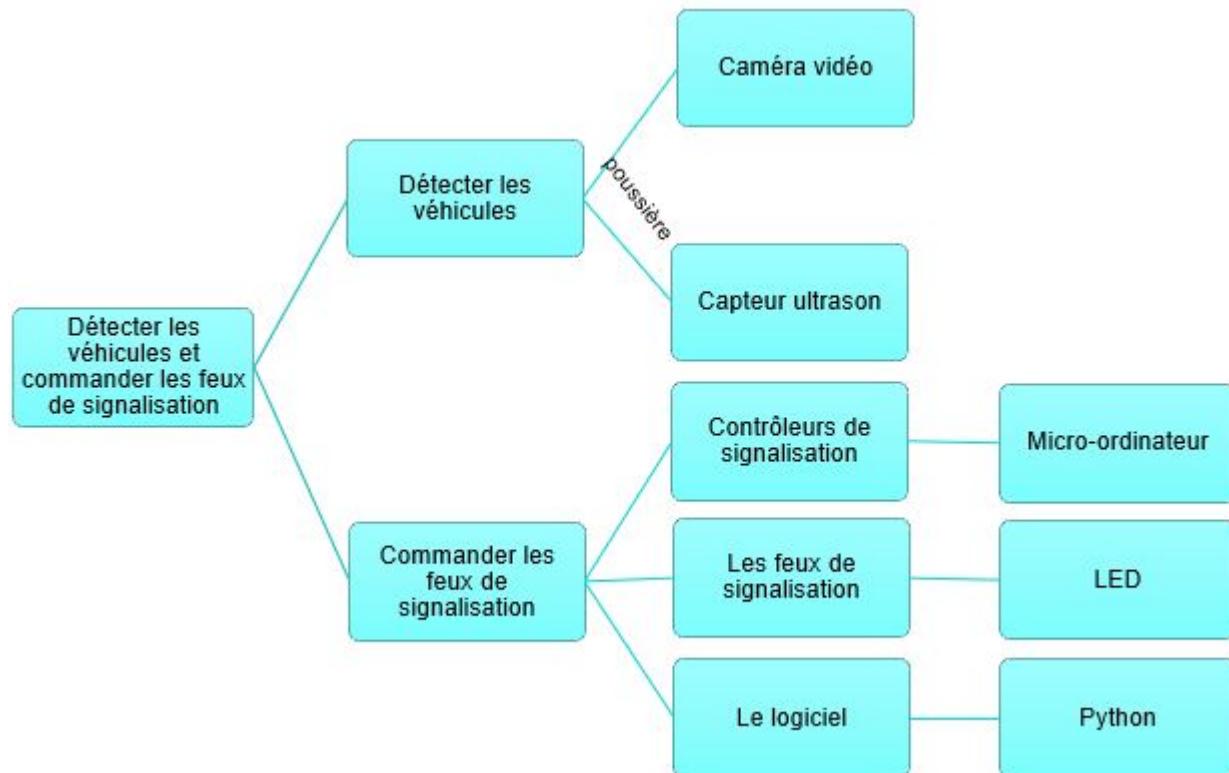


Figure 20 : Diagramme FAST du système

0.6.6 Réalisation et tests

La réalisation de notre projet s'est articulée autour de deux aspects essentiels : la partie logicielle (software) et la partie matérielle (hardware). Ces deux composantes ont été développées en parallèle pour assurer le bon fonctionnement et l'intégration harmonieuse de notre solution de gestion des embouteillages.

Nous essayeront à travers cette partie du rapport développer ces deux aspects et décrire les différentes étapes d'implémentation de chaque aspect.

0.4.6.1 L'aspect logiciel (Software)

Dans la partie logicielle, nous avons consacré nos efforts à la conception et au développement des différents algorithmes et méthodes du machine-learning qui permettent d'analyser les données capturées par les capteurs et les caméras, de prendre des décisions en temps réel et d'ajuster de manière adaptative la durée des feux de signalisation en fonction du trafic.

Pour implémenter ceci, nous avons utilisé le langage de programmation Python, reconnu pour sa performance en termes de traitement des données et d'images.

0.4.6.1.1 Le code de détection des voitures

Afin de détecter les voitures présentes dans les photos fournies par la caméra, nous avons recours à des méthodes de machine-learning, et en particulier nous avons exploité le modèle YOLO (You Only Look Once) qui est un modèle de détection d'objets en temps réel.

Puisque nous cherchons éventuellement à mettre en œuvre un prototype qui démontre le bon fonctionnement de notre projet, nous avons donc opté pour l'utilisation de voiture de jeu pour simuler les voitures présentes dans un croisement, cependant, la démarche utilisée pour la détection reste la même pour des voitures réelles.



Figure 21 : Les voitures de jeu utilisées dans notre prototype

Néanmoins, nous avons rencontré un premier obstacle qui est l'indisponibilité d'une Dataset open-source afin d'entraîner notre modèle de détection des voitures de jeu.

Nous étions donc obligés de créer notre propre Dataset qui nous permettra d'entraîner notre modèle YOLO. Pour cela, nous avons utilisé **Roboflow**, un fournisseur d'outils de développement d'applications de vision par ordinateur.



Figure 22 : l'outil ROBOFLOW utilisé pour la création du dataset

La création du jeu de données passe par les étapes suivantes :

- On commence d'abord par prendre des centaines de photos des voitures de jeu qu'on utilisera dans notre prototype.



Figure 23 : Exemple 1 d'une photo prise des voiture de jeu pour être utilisée comme élément du dataset



Figure 24 : Exemple 2 d'une photo prise des voitures de jeu pour être utilisée comme élément du dataset

- Ensuite vient l'étape **d'annotation** où on doit annoter une par une les photos prises des voitures de jeu. L'annotation est quand on ajoute des "bounding boxes" autour des voitures prises en photo.

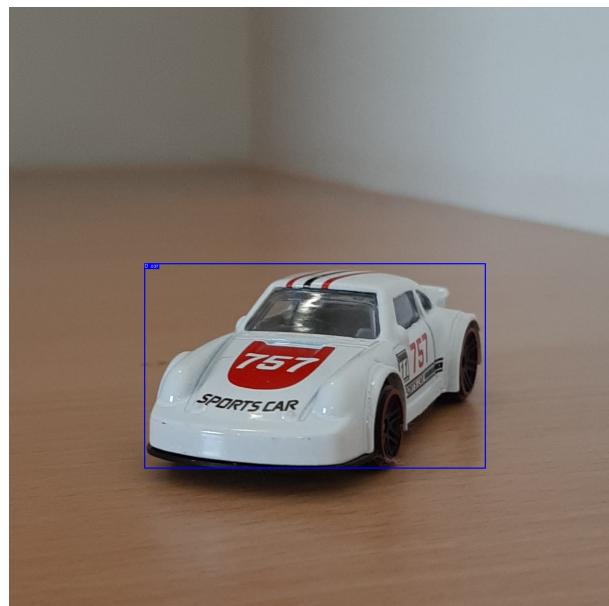


Figure 25 : Exemple 1 de l'annotation d'une photo de voiture utilisée dans la dataset.

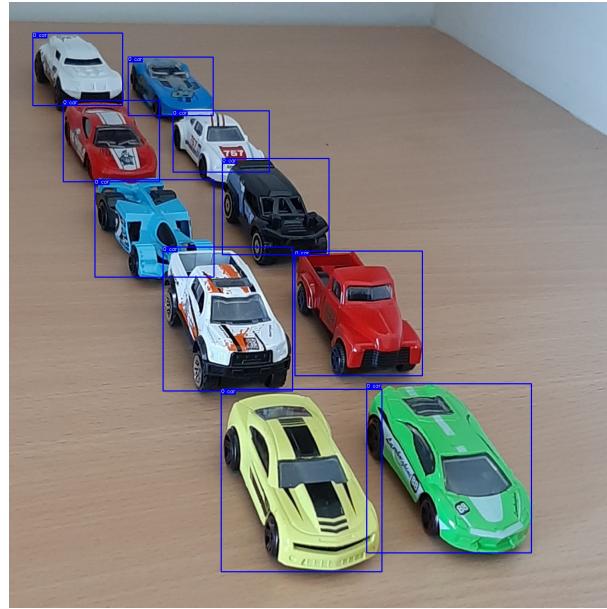


Figure 26 : Exemple 2 de l'annotation d'une photo d'un ensemble de voitures utilisée dans la dataset

- Par la suite vient l'étape de l'**augmentation des données (Data augmentation)** qui permet d'améliorer la précision de notre modèle.

La Data augmentation est un processus qui nous permet de générer des données supplémentaires d'entraînement à travers plusieurs techniques (Data augmentation techniques) qui varient en fonction du type des données considérées. Parmi ces techniques on trouve : la rotation, la translation, le cadrage (cropping) pour les images, l'ajout du bruit (noise injection), le shifting, etc, pour les audios.

4

Augmentation

② What can augmentation do?

Create new training examples for your model to learn from by generating augmented versions of each image in your training set.

90° Rotate Clockwise, Counter-Clockwise	Edit	x
Crop 19% Minimum Zoom, 39% Maximum Zoom	Edit	x
Mosaic	Edit	x
Bounding Box: Rotation Between -11° and +11°	Edit	x
Bounding Box: Shear ±15° Horizontal, ±15° Vertical	Edit	x
+ Add Augmentation Step		

Figure 27 : les techniques de Data augmentation utilisée dans notre projet

s

- Enfin, nous divisons notre Dataset créé en trois ensemble, le **training set**, le **validation set** et **enfin le test set**.

Le training Dataset sera utilisé par la suite pour entraîner notre modèle de détection des voitures. La phase d'entraînement du modèle de détection a nécessité un investissement considérable en termes de temps, étant donné que nous étions encore en phase de découverte et d'apprentissage de la manipulation des ensembles de données (Datasets), cependant, on est arrivé à avoir des résultats satisfaisants. On est arrivé à atteindre une précision moyenne (Mean average precision mAP) de 99.5%, une précision de 99% et un rappel (recall) de 99.5%

En effet, la mAp, la précision et le rappel sont des métriques utilisées pour évaluer la précision des modèles de détection.

La mAP (mean Average Precision) est une mesure couramment utilisée pour évaluer la performance des modèles de détection d'objets. Elle est calculée en moyennant les précisions moyennes pour différentes valeurs de seuils de similarité (IoU - Intersection over Union) entre les prédictions du modèle et les vérités terrain. La mAP fournit une évaluation globale de la précision du modèle sur l'ensemble des classes d'objets.

Alors que la précision (precision) est une mesure qui évalue la proportion d'instances prédites comme positives qui sont réellement positives. Elle représente la capacité du modèle à identifier correctement les véritables positifs et à minimiser les faux positifs (erreur de type 1). Une précision élevée indique que le modèle produit relativement peu de prédictions erronées.

Le rappel (recall), quant à lui, est une mesure qui évalue la proportion d'instances positives réellement détectées par le modèle. Il représente la capacité du modèle à identifier tous les vrais positifs et à minimiser les faux négatifs. Un rappel élevé indique que le modèle est capable de détecter la majorité des instances positives présentes dans les données.

En résumé, la mAP permet d'évaluer la performance globale d'un modèle de détection d'objets, tandis que la précision et le rappel fournissent des informations plus spécifiques sur la capacité du modèle à effectuer des prédictions précises et à détecter les véritables positifs.

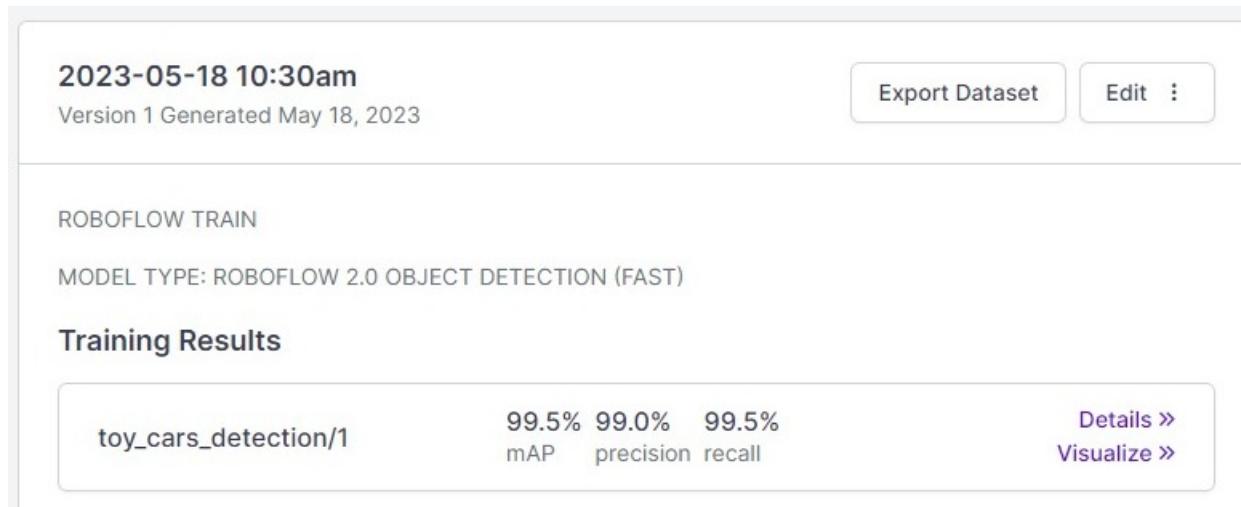


Figure 28 : Les résultats obtenus après la phase d'entraînement du modèle

Enfin, après l'entraînement du modèle, celui-ci est prêt à être utilisé pour la détection des voitures en temps réel. Ci-après le code utilisé pour la détection des voitures et son application sur une photo qui ne figure pas dans le training set et que le modèle n'a jamais traité avant.

```
[3] from roboflow import Roboflow
rf = Roboflow(api_key="D3RwR8TzDH3uzPjmMdC")
project = rf.workspace().project("toy_cars_detection")
model = project.version(1).model

# infer on a local image
print(model.predict("path to car image", confidence=40, overlap=30).json())

# visualize your prediction
model.predict("path to car image", confidence=40, overlap=30).save("image.jpg")
```

Figure 29 : Le code python qui permet d'utiliser le modèle entraîné sur de nouvelles photos.



Figure 30 : première photo utilisée pour tester le modèle entraîné.

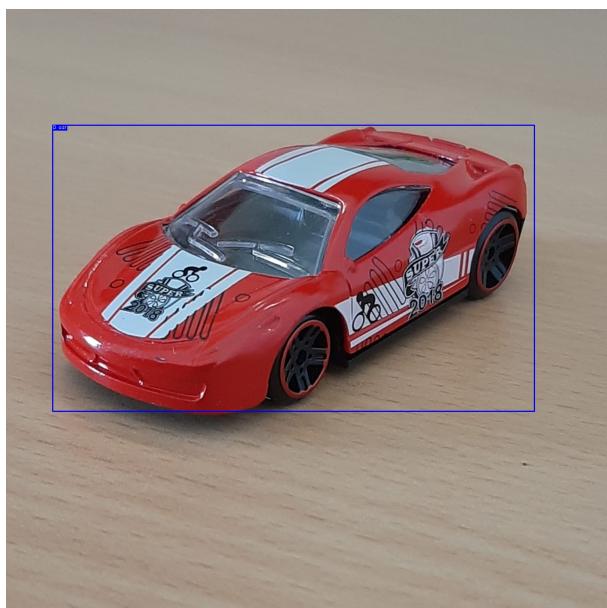


Figure 31 : output du code de détection appliqué à la photo précédente où on voit clairement que le modèle était capable de détecter la voiture dans la photo.



Figure 32 : deuxième photo utilisée pour tester le modèle entraîné.

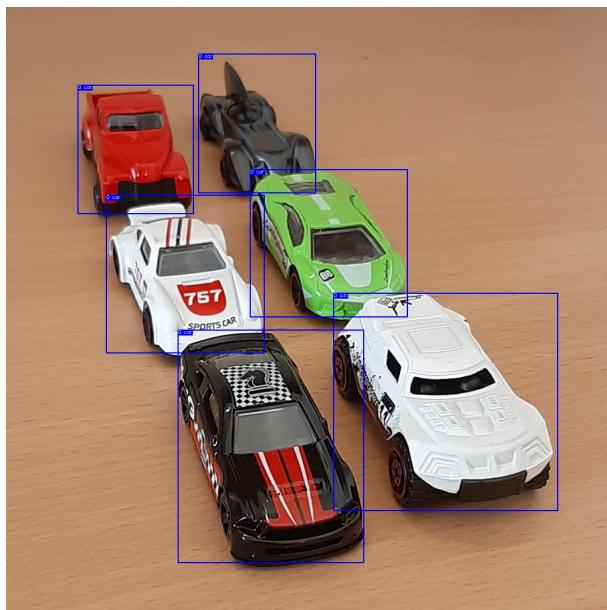


Figure 33 : output du code de détection appliqué à la photo précédente où on voit clairement que le modèle était capable de détecter les voitures dans la photo.

0.4.6.1.2 Le code de décompte du nombre des voitures

Après avoir réussi à détecter les voitures dans une photo donnée, vient maintenant l'étape de compter le nombre de voitures détectées. Pour cela, nous avons utilisé la librairie cv2 de python. Ci-après le code de décompte et son application à des photos de voitures :

```

✓ [13] #Car counting
import cv2
import numpy as np

def count_blue_boxes(image_paths):
    num_boxes = 0

    for image_path in image_paths:
        # Load the image
        image = cv2.imread(image_path)

        # Convert the image to the HSV color space
        hsv = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)

        # Define the lower and upper blue color range
        lower_blue = np.array([120, 50, 50]) #kant flwl (90, 50, 50)
        upper_blue = np.array([130, 255, 255])

        # Create a mask for the blue color range
        mask = cv2.inRange(hsv, lower_blue, upper_blue)

        # Apply morphological operations to remove noise
        kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5, 5))
        mask = cv2.morphologyEx(mask, cv2.MORPH_OPEN, kernel)

        # Find contours in the mask
        contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

        # Filter contours based on area and aspect ratio
        min_area = 500 # Minimum contour area to consider
        max_aspect_ratio = 1.5 # Maximum aspect ratio of a valid box

        for contour in contours:
            area = cv2.contourArea(contour)
            x, y, w, h = cv2.boundingRect(contour)
            aspect_ratio = w / float(h)

            if area > min_area and aspect_ratio < max_aspect_ratio:
                num_boxes += 1

    return num_boxes

```

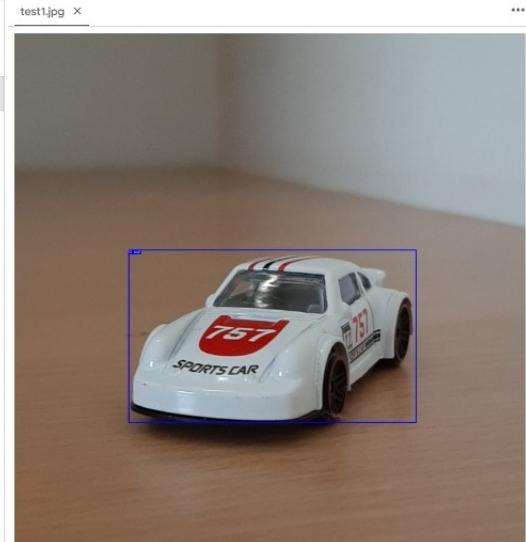


Figure 34 : La première partie du code de décompte appliquée à une photo de voiture de jeu.

```

# Create a mask for the blue color range
mask = cv2.inRange(hsv, lower_blue, upper_blue)

# Apply morphological operations to remove noise
kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5, 5))
mask = cv2.morphologyEx(mask, cv2.MORPH_OPEN, kernel)

# Find contours in the mask
contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

# Filter contours based on area and aspect ratio
min_area = 500 # Minimum contour area to consider
max_aspect_ratio = 1.5 # Maximum aspect ratio of a valid box

for contour in contours:
    area = cv2.contourArea(contour)
    x, y, w, h = cv2.boundingRect(contour)
    aspect_ratio = w / float(h)

    if area > min_area and aspect_ratio < max_aspect_ratio:
        num_boxes += 1

return num_boxes

# Example usage
image_paths = [
    "/content/drive/MyDrive/PLBO/test1.jpg",
    # Add more image paths as needed
]

num_blue_boxes = count_blue_boxes(image_paths)
print(f"number of cars: {num_blue_boxes//2 + 1}")

```



Figure 35 : La deuxième partie du code de décompte appliquée à une photo de voiture de jeu, le résultat, affiché en bas, montre que le programme arrive à compter le nombre de voiture dans la photo.

```

  # Create a mask for the blue color range
  mask = cv2.inRange(hsv, lower_blue, upper_blue)

  # Apply morphological operations to remove noise
  kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5, 5))
  mask = cv2.morphologyEx(mask, cv2.MORPH_OPEN, kernel)

  # Find contours in the mask
  contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

  # Filter contours based on area and aspect ratio
  min_area = 500 # Minimum contour area to consider
  max_aspect_ratio = 1.5 # Maximum aspect ratio of a valid box

  for contour in contours:
    area = cv2.contourArea(contour)
    x, y, w, h = cv2.boundingRect(contour)
    aspect_ratio = w / float(h)

    if area > min_area and aspect_ratio < max_aspect_ratio:
      num_boxes += 1

  return num_boxes

# Example usage
image_paths = [
  "/content/drive/MyDrive/PLBD/test7.jpg",
  # Add more image paths as needed
]

num_blue_boxes = count_blue_boxes(image_paths)
print(f"number of cars: {num_blue_boxes//2 +1}")

number of cars: 5

```

Figure 36 : Un deuxième test du code sur une deuxième photo, on voit aussi que le programme arrive à compter le nombre de voitures dans la photo.

0.4.6.1.3 Le code final

Enfin, une fois les codes de détection et de décompte développés, nous avons procédé à l'écriture du code final qui assure le contrôle des feux de signalisation en fonction du nombre de voitures détectées. Ce code central permet également de gérer les différentes composantes de notre solution, telles que le capteur de poussière, le capteur ultrason, et autres. Ci-après le code central :

0.4.6.2 L'aspect matériel (Hardware)

Dans la partie matérielle de notre projet, nous avons consacré nos efforts à la fabrication et à l'assemblage des différentes composantes physiques nécessaires à notre solution. Nous avons pris en charge la conception et la construction des éléments tels que les planches de bois, les tiges, et autres éléments structurels, en utilisant le matériel disponible au FABLAB

Parallèlement à nos travaux sur la partie logicielle, cette étape nous a permis de concrétiser notre vision globale du projet en alignant l'aspect matériel avec les algorithmes et méthodes développés pour l'analyse des données et la gestion du trafic.

0.4.6.2.1 La fabrication du support du prototype

Dans le cadre de la fabrication de notre prototype, une étape essentielle a été la conception et la construction du support qui accueille l'ensemble de notre système. Pour cela, nous avons opté principalement pour l'utilisation de planches de bois, en raison de leur disponibilité, de leur robustesse et de leur facilité de manipulation.

La première étape a consisté à prendre des mesures précises et à élaborer un plan détaillé pour assurer la stabilité et la solidité du support. Ensuite, nous avons procédé à la découpe des planches selon les dimensions spécifiées dans le plan, en veillant à obtenir des pièces parfaitement ajustées. Pour cela, nous avons utilisé la découpe laser disponible dans la FABLAB.

Nous avons découpé deux planches de bois, une qui portera le système embarqué de notre solution, à savoir la carte Raspberry pi, et une autre planche qui portera la maquette de notre solution. Nous avons

utilisé des techniques de fixation telles que le vissage pour assembler les différentes parties du support, en veillant à ce qu'il soit solide et stable.

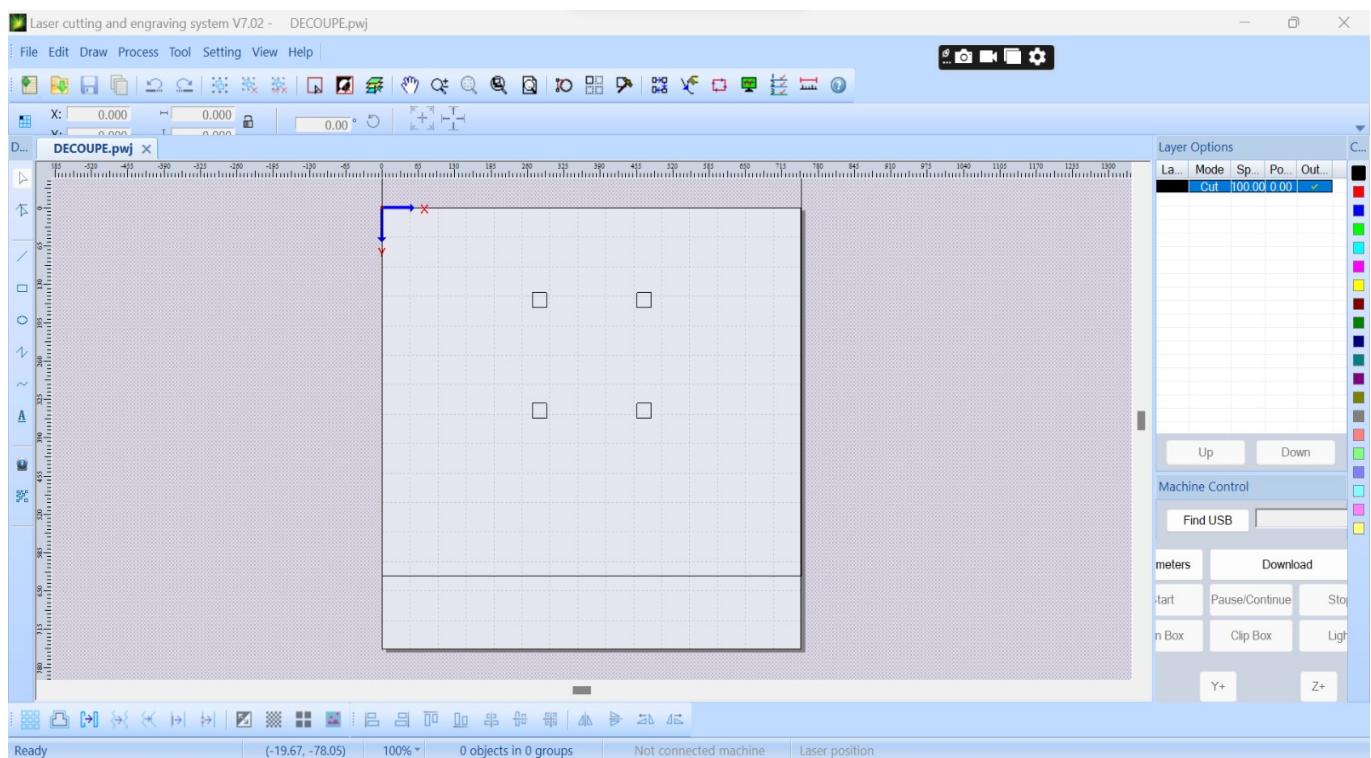


Figure 38 : Le plan des planches élaboré par le logiciel Laser cutting and engraving system



Figure 39 : Photo des deux planches de bois coupées par la découpe laser.

0.4.6.2.2 La fabrication du support des feux de signalisation

Après avoir réalisé la conception et le montage du support principal de notre prototype, nous avons entamé la deuxième étape de fabrication des éléments matériels de notre solution. Cette étape consistait spécifiquement à fabriquer les tiges qui serviraient de supports pour les feux de signalisation et les caméras-vidéo. Les tiges ont été soigneusement conçues et fabriquées pour assurer la stabilité et la résistance nécessaires, tout en offrant un positionnement optimal des équipements. Ce processus de fabrication nous a permis de créer des supports solides et fonctionnels, intégrant harmonieusement les différents composants de notre système.

La conception des tiges a été réalisée à l'aide du logiciel "Laser cutting and engraving system", qui nous a permis de créer des modèles précis et personnalisés. Nous avons utilisé du plexiglas comme matériau de fabrication en raison de ses qualités esthétiques, de sa durabilité et de sa facilité de manipulation. Le plexiglas a été découpé selon les dimensions spécifiées par le logiciel, en utilisant un processus de découpe au laser pour obtenir des résultats précis et de haute qualité.



Figure 40 : Photo des tiges fixées sur le support principal du prototype

0.6.7 Résultats obtenus : Analyse et commentaire

Une fois notre solution mise en œuvre et les deux aspects essentiels, à savoir l'aspect logiciel et l'aspect matériel, assemblés, nous avons réussi à créer un système de feux de signalisation adaptatifs pleinement opérationnel. Ce système est capable de gérer efficacement le trafic routier dans les intersections de la ville de Dakar.

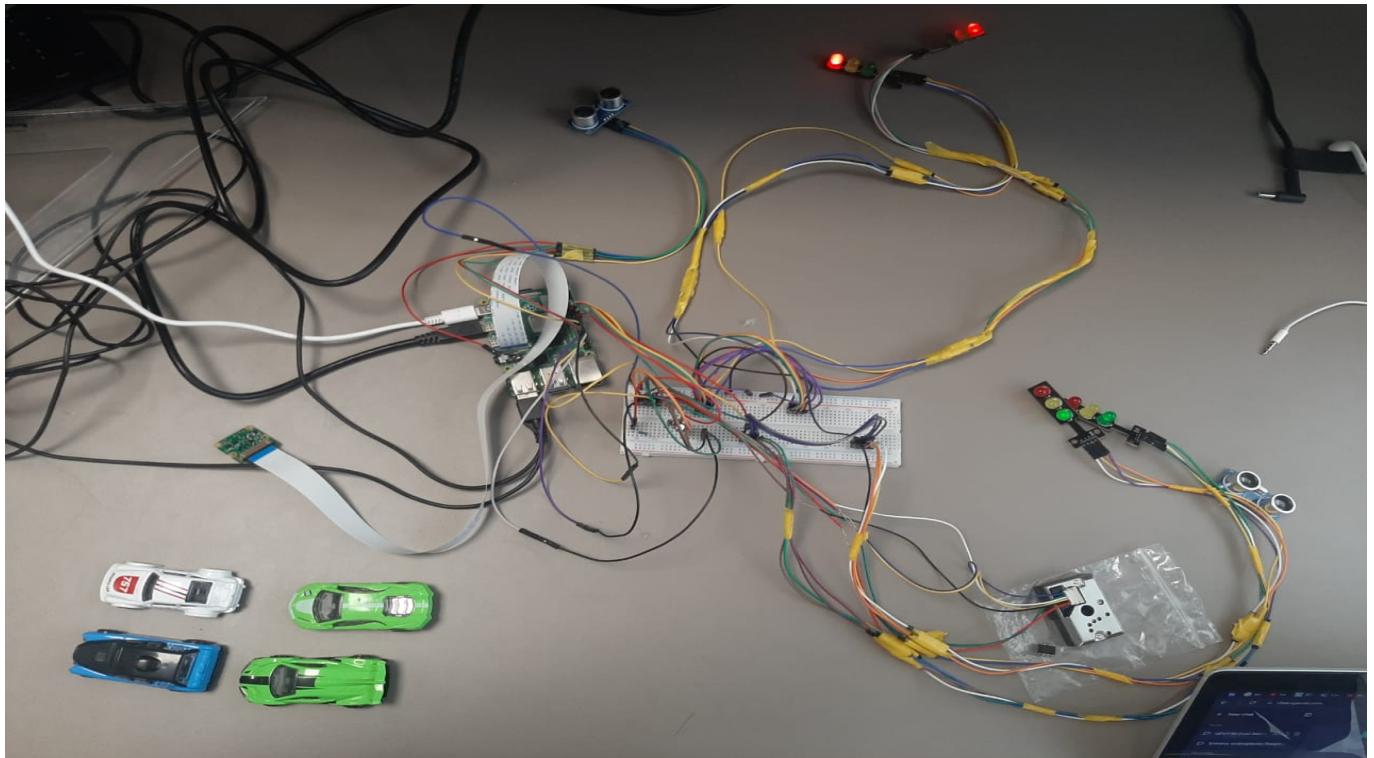


Figure 41 : Assemblage des différentes composantes de la solution

La première phase d'assemblage nous a offert l'opportunité de tester le bon fonctionnement de notre solution et d'obtenir des résultats préliminaires. Ces résultats se sont avérés satisfaisants, confirmant ainsi l'efficacité de notre système.

Ensuite, nous avons entamé l'étape finale du montage du prototype, en veillant à respecter des normes de solidité et d'esthétique. Cette phase était cruciale pour garantir que notre solution soit à la fois robuste et esthétiquement agréable, prête à être exposée le jour de la foire.

Nous sommes fiers d'avoir pu développer et concrétiser notre solution, qui se distingue par sa prise en compte et son adaptation aux conditions climatiques particulières de la ville de Dakar. Cette caractéristique constitue une originalité majeure de notre projet, car elle garantit l'efficacité et la fiabilité de notre système, même dans des situations météorologiques spéciales telles que les tempêtes de poussière.

0.7 Conclusion générale

En conclusion, notre projet a été axé sur la résolution du problème des embouteillages routiers dans les carrefours de la ville de Dakar, en mettant en place un système de feux de signalisation adap-

tatifs. Nous avons accompli plusieurs étapes clés, allant de la recherche initiale sur le contexte et la problématique, à la conception et au développement du système, en passant par la collecte de données, l'entraînement du modèle de détection, et la fabrication du support matériel.

Les principales contributions de notre projet résident dans la mise en place d'un système de feux de signalisation adaptatifs, capable de gérer intelligemment le trafic routier en ajustant la durée des feux en fonction du nombre de voitures présentes dans chaque voie. Nous avons obtenu des résultats satisfaisants, avec une précision moyenne de détection des voitures de 99,5% et une précision globale de 99%.

Cependant, notre projet n'a pas été sans difficultés. La phase d'entraînement du modèle de détection a demandé du temps et de l'effort, notamment en raison de notre non familiarisation avec la manipulation des ensembles de données. De plus, la prise en compte des conditions particulières de Dakar, telles que les tempêtes de poussière, a nécessité l'intégration de capteurs spécifiques pour garantir le bon fonctionnement du système.

Ainsi, notre projet a permis de proposer une solution pour atténuer les embouteillages routiers à Dakar. Les résultats obtenus sont encourageants et ouvrent la voie à de nouvelles perspectives pour améliorer la mobilité urbaine dans les grandes villes africaines. Nous avons tiré de précieux enseignements de cette expérience comme l'importance de la planification et l'anticipation dans la réussite d'un projet, la primordialité du travail en équipe et de la bonne communication pour le bon déroulement des différentes étapes du projet.

0.8 Références bibliographiques

- [1] : S. Writer, « How much time South Africans lose to traffic every day »
<https://businessstech.co.za/news/motoring/170661/how-much-time-south-africans-lose-to-traffic-every-day/> (consulté le 12 juin 2023)
- [2] : O. L. R. Akiyo, J. Adounkpe, C. L. Babajide, I. Akobi, et B. N'bessa, « L'embouteillage Dans Les Grandes Villes de L'Afrique de L'ouest et Ses Problemes : Cas de Cotonou Au Benin.” », CODES-RIA, vol.20, n 1, p. 87-110, 2016.
- [3] : « Le «smart» pour relever le défi de la mobilité urbaine », La Tribune.
<https://afrique.latribune.fr/africa-tech/smart-cities/2019-11-26/le-smart-pour-relever-le-defi-de-la-mobilite-urbaine-833355.html> (consulté le 15 juin 2023).
- [4] : « Côte d'Ivoire : la mobilité urbaine, un réel casse-tête à Abidjan à cause des embouteillages monstrueux », Le360 Afrique. <https://afrique.le360.ma/cote-divoire/societe/2022/01/22/36745-cotedivoire-la-mobilite-urbaine-un-reel-casse-tete-abidjan-cause-des-embouteillages> (consulté le 15 juin 2023).
- [5] : C. Laganier, « La décongestion intelligente, ou bientôt la fin des embouteillages ? », EnergyStream, 13 juin 2016. <https://www.energystream-wavestone.com/2016/06/decongestion-intelligente-bientot-fin-embouteillages/> (consulté le 16 juin 2023).
- [6] : « Des feux de circulation “intelligents” pour réduire la congestion Radio-Canada.ca ». <https://ici.radiocanada.ca/nouvelle/1069314/des-feux-de-circulation-intelligents-pour-reduire-la-congestion> (consulté le 16 juin 2023)
- [7] : « Les péages urbains ».
https://www.citego.org/bdf_fiche-document-198fr.html (consulté le 16 juin 2023).
- [8] : « A-t-on des solutions pour en finir avec les bouchons de circulation ? — Le Devoir ». <https://www.ledevoir.com/politique/regions/509885/pour-en-finir-avec-les-bouchons-des-solutions> (consulté le 17 juin 2023).
- [9] : « Les avantages des voitures autonomes ». <https://www.ecoleauto.com/blog/les-avantages-des-voituresautonomes/> (consulté le 17 juin 2023)
- [10] : « Des chercheurs misent sur la conduite autonome pour réduire les bouchons sur l'autoroute ». <https://www.usinedigitale.fr/article/des-chercheurs-misent-sur-la-conduite-autonome-pour-reduire-les-bouchons-sur-lautoroute.N2071896> (consulté le 17 juin 2023).
- [11] : « Voies lyonnaises : plus de bouchons sur le quai Claude-Bernard ? Les écologistes répondent — Actu Lyon ». https://actu.fr/auvergne-rhone-alpes/lyon_69123/voies-lyonnaises-plus-de-bouchons-sur-le-quai-claude-bernard-les-ecologistes-repondent_49308262.html (consulté le 17 juin 2023)
- [12] : Eradius Edward Rwakarehe, « Review of Strategies for Curbing Traffic Congestion in Sub-Saharan Africa Cities : Technical and Policy Perspectives », vol. 40, p. 24-32, déc. 2021.
- [13] : Xochipilli, « Le paradoxe de Braess – Le Webinet des Curiosités ». <https://webinet.cafe->

sciences.org/articles/leparadoxe-de-braess/ (consulté le 17 juin 2023).

[14] : Eradius Edward Rwakarehe, « Review of Strategies for Curbing Traffic Congestion in Sub-Saharan Africa Cities : Technical and Policy Perspectives », vol. 40, p. 24-32, déc. 2021.

[15] : « Pénurie de compétences en Afrique du Sud [Business Africa] — Africanews ». <https://fr.africanews.com/2022/02/06/penurie-de-competences-en-afrique-du-sud-business-africa/> (consulté le 18 juin 2023).

[16] : « Les embouteillages : le principal pari de la mobilité urbaine à Dakar – Askan ». <https://askan.co/2022/04/lesembouteillages-le-principal-pari-de-la-mobilite-urbaine-a-dakar/> (consulté le 18 juin 2023).

[17] : « Transport ferroviaire : 120 milliards de FCfa de la Bad pour le projet Ter — Ministère des Infrastructures, des Transports terrestres et du Déseinclavement - MITTD ». <https://www.mittd.gouv.sn/fr/content/transport-ferroviaire-120-milliards-de-fcfa-de-la-bad-pour-le-projet-ter> (consulté le 18 juin 2023).

[18] : « Le BRT en chiffres : 300 milliards FCFA, 144 bus, 18,3 km, 300 mille voyageurs/jour », impact.sn : informations, enquêtes, analyses. https://www.impact.sn/Le-BRT-en-chiffres-300-milliards-FCFA-144-bus-183-km-300-millevoyageurs-jour_a16258.html (consulté le 18 juin 2023).

[19] : « INCROYABLE ! Les embouteillages à Dakar coûtent 108 milliards FCFA par an (ministre) - SAMAREW INFOS ». <https://samarew.com/incroyable-%e2%80%8bles-embouteillages-a-dakar-coutent-108-milliards-fcfa-par-an-ministre/> (consulté le 18 juin 2023).

[20] : L. Sowmya, « Adaptive Traffic Control System », efftronics. <https://www.efftronics.com/adaptive-traffic-controlsystem> (consulté le 18 juin 2023)