



Filière : Electronique, Energie Electrique et Automatique

Spécialité : M2 Systèmes Embarqués et Traitement de l'Information (SETI)

Rapport de Projet Proto204

Road Traffic Monitoring Based on Integration of IoT and Machine Learning

Travail réalisé par :

▪ Yassine OUZAR

Encadré par :

▪ M. Ronan JAMES

Année Universitaire : 2018 / 2019

Introduction et problématique :

Vu l'augmentation des véhicules circulants et le problème d'embouteillage qu'on rencontre souvent sur les différentes routes notamment dans les grandes villes, il est devenu nécessaire de trouver des solutions à ce problème pour garantir la fluidité du trafic routier et diminuer le temps perdu sur les trajets à cause des perturbation de la circulation routière.

Les solutions existantes sont soit coûteuses et complexes à mettre en œuvre tels que les systèmes qui utilisent la technologie V2X, ou bien les systèmes cartographiques comme l'application waze et v-traffic qui sont basés sur les données des utilisateurs pour fournir des informations sur le trafic routier.

Les récents travaux de recherche utilisent les méthodes de détection des objets dans les vidéos pour analyser la circulation routière afin de trouver des solutions pour diminuer les embouteillages. Dans ce contexte, ma solution que je propose utilise un modèle pré-entraîné pour la détection et comptage des véhicules dans une scène vidéo et afficher les résultats sur une interface web.

Les objectifs :

- Intégrer les méthodes du machine learning dédiées à l'embarqué pour détecter les véhicules dans les vidéos.
- Utiliser des outils open source (opencv, tensorflow, node.js ...) et un matériel peu coûteux (Raspberry PI+caméra)
- Surveiller l'état du trafic en temps réel.

Solution technique :

Implémenter un système de détection des véhicules dans une scène vidéo capable de surveiller le trafic routier et fournir en temps réel des informations sur le comportement des véhicules et les piétons sur les différentes routes via une interface web.

Partie 1 : Détection et comptage des véhicules



Figure 1 : Les outils utilisés pour la détection et comptage des véhicules.

1. Capture du flux vidéo et récupérer à l'aide de Numpy le cadre de l'image, pour l'afficher ensuite sur l'écran.
2. Détection en temps réels les objets qui figurent sur les images capturées à l'aide du modèle d'apprentissage déjà défini appelé « SSDlite_mobilenet_v2_coco_2018_05_09 » (modèle de

réseaux de neurone convolutifs extrêmement léger et petits en terme de lignes de code et les poids des modèles) dédié à l'embarqué et qui permet de reconnaître 90 objets différents.

3. Classifier les objets détectés selon leurs types et extraire la classe des véhicules.
4. Compter le nombre des véhicules détectés.
5. Afficher le nombre des véhicules détectés sur le flux vidéo de sortie.

Partie 2 : Surveillance à distance de l'état du trafic via l'interface web

L'architecture logicielle du système est de type client/serveur.

Partie client : représente l'interface web qui permet aux utilisateurs distants (conducteurs, responsables de gestion du trafic) d'accéder à la plateforme Raspberry PI, lancer le programme de détection et comptage des véhicules pour surveiller l'état du trafic routier. Elle est réalisée principalement en HTML, CSS et JavaScript.

Partie serveur : La partie serveur ou back-end de mon projet est fondée sur une plateforme de développement en JavaScript appelée node.js permettant de mettre en œuvre des applications réseaux asynchrones et une communication en temps réel. Le serveur reçoit les requêtes http envoyées par l'utilisateur distant, les traite et l'exécute en lançant les différents programmes et afficher les résultats sur la page web.



Figure 2 : Les technologies web utilisées pour implémenter l'interface web.

Road Traffic Monitoring



Launch

Count: Nombre de Véhicules Détectés:

Figure 3 : L'interface web du projet.

Conclusion et perspectives :

Contrairement aux systèmes de gestion de trafic existants, ma solution proposée dans ce projet permet aux responsables de gestion de trafic de surveiller en temps réel l'état du trafic et les périodes d'embouteillages afin de trouver des solutions pour diminuer les bouchons sans avoir besoin d'utiliser les données fournies par les utilisateurs.

En ajoutant plusieurs fonctionnalités, mon système peut concurrencer les systèmes existants de surveillance du trafic routier.

En guise de perspectives, un complément de ce travail peut être réalisé en utilisant des bibliothèques JavaScript telles que Echarts ou Morris.js pour transformer le nombre de véhicules détectés au cours du temps à une représentation graphique des statistiques permettant d'estimer les périodes de bouchon.

Les fonctionnalités à ajouter pour améliorer mon projet :

- Indiquer l'état du trafic en temps réel
- Prédire les périodes d'embouteillages
- Fournir des itinéraires plus rapides

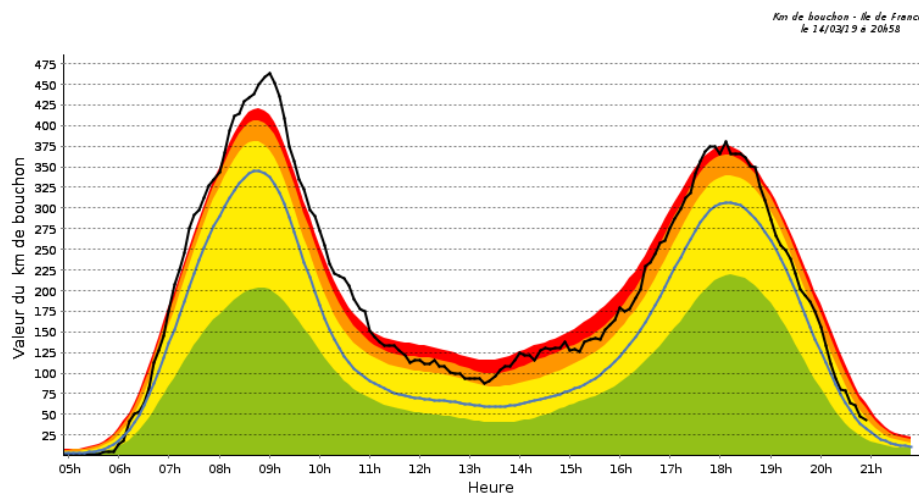


Figure 4 : Statistiques sur les périodes d'embouteillages.