



Fiche de lecture

Les réseaux de neurones convolutifs dans le monde des cylindrées

Auteur
GHOUIBI Ghassen

Encadré par
Mr. Jean Jacques Mariage

Contents

I Introduction	2
I.1. Contexte	2
I.2. Problématique	2
II CNN	3
II.1. Définition	3
II.2. Algorithme	4
III Solution CNN	4
III.1. Détection de signalisation	4
III.2. Détection de collision	6
III.3. Détection de dommages matériel	7
III.4. Voiture autonome	7
IV Mon avis	8
V Conclusion	8

I Introduction

Les tâches que les humains pensent qu'il sont acquises sont souvent les plus difficiles à accomplir pour les machines.

C'est pourquoi quand on vous demande de prouver votre humanité à travers ces tests CAPTCHA, on vous pose toujours une question ridicule et simple, par exemple si une image contient ou non un panneau routier.

Ces tests sont efficaces pour déterminer si un utilisateur est humain parce que la reconnaissance d'image en contexte est difficile pour les machines.

Les ordinateurs pour répondre avec précision à ce type de questions de manière automatisée et efficace pour de grandes quantités de données est compliquée.

Pour répondre à ce problème, des entreprises comme Amazon dépensent beaucoup d'argent pour traiter manuellement les problèmes de classification des images et des vidéos.

L'étiquetage manuel des données est quelque chose que nous voyons déjà dans des entreprises comme Facebook pour organiser des histoires de flux d'actualités.

L'examen manuel de millions d'images et de vidéos pour identifier certains types de contenu est extrêmement fastidieux et coûteux.

Pourtant, malgré cela il existe peu de techniques pour analyser efficacement le contenu d'images et de vidéos de manière automatisée.

I.1 Contexte

La classification avec des réseaux de neurones artificiels est une approche très populaire pour résoudre les problèmes de reconnaissance de formes.

Un réseau de neurones est un modèle mathématique basé sur la connexion à travers l'autre unités neuronales de la même manière que les réseaux neuronaux biologiques.

Dans notre présentation on va étudier de plus proche les réseaux de neurones convolutifs sur un sujet important c'est les accidents en voiture.

Les accidents de voitures en France provoque 9 morts par jour !

La distraction est une cause majeure des accidents de la route en tête de liste on trouve qu'envoyer des messages ou téléphoner peut augmenter vos chance que faire un accident fatal bien sûr aligné avec l'alcool au même degré.

I.2 Problématique

Est ce que les nouvelles solutions trouvés dans le domaine de classification d'image, machine learning et prédiction peuvent réduire les accidents de la route voir les éliminer complètement ?

II CNN

II.1 Définition

Les réseaux de neurones convolutifs ont une méthodologie similaire à celle des méthodes traditionnelles d'apprentissage supervisé : ils reçoivent des images en entrée, détectent les features de chacune d'entre elles, puis entraînent un classifieur dessus.

Les principaux éléments sont les suivants :

- Un réseau de neurones est un système composé de neurones, généralement répartis en plusieurs couches connectées entre elles
- Un tel système s'utilise pour résoudre divers problèmes statistiques, mais nous nous intéressons ici qu'au problème de classification (très courant).
Dans ce cas, le réseau calcule à partir de l'entrée un score (ou probabilité) pour chaque classe.
La classe attribuée à l'objet en entrée correspond à celle de score le plus élevé
- Chaque couche reçoit en entrée des données et les renvoie transformées. Pour cela, elle calcule une combinaison linéaire puis applique éventuellement une fonction non-linéaire, appelée fonction d'activation.
Les coefficients de la combinaison linéaire définissent les paramètres (ou poids) de la couche
- Un réseau de neurones est construit en empilant les couches : la sortie d'une couche correspond à l'entrée de la suivante.
- Cet empilement de couches définit la sortie finale du réseau comme le résultat d'une fonction différentiable de l'entrée
- La dernière couche calcule les probabilités finales en utilisant pour fonction d'activation la fonction logistique (classification binaire) ou la fonction softmax (classification multi-classes)
- Une fonction de perte (loss function) est associée à la couche finale pour calculer l'erreur de classification, Il s'agit en général de l'entropie croisée.
- Les valeurs des poids des couches sont appris par rétropropagation du gradient : on calcule progressivement (pour chaque couche, en partant de la fin du réseau) les paramètres qui minimisent la fonction de perte régularisée.
L'optimisation se fait avec une descente du gradient stochastique

II.2 Algorithme

Une architecture de réseau de neurones convolutifs est formée par un empilement de couches de traitement :

- la couche de convolution (CONV) qui traite les données d'un champ récepteur
- la couche de pooling (POOL), qui permet de compresser l'information en réduisant la taille de l'image intermédiaire (souvent par sous-échantillonnage)
- la couche de correction (ReLU), souvent appelée par abus « ReLU » en référence à la fonction d'activation (Unité de rectification linéaire)
- la couche « entièrement connectée » (FC), qui est une couche de type perceptron
- la couche de perte (LOSS).

Voici une fonction du training de CNN:

Algorithm 3. Pseudo code of CNNTRAIN

Input: cnn.layers={'I', 'C', 'S', 'C', 'S'};
Output: cnn.layers={'I', 'C', 'S', 'C', 'S'};

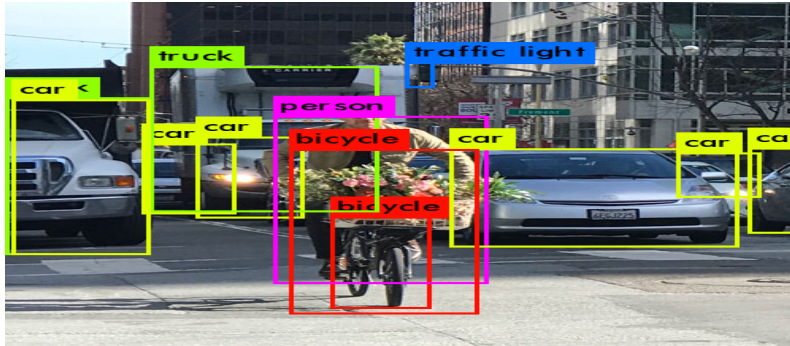
1: **Calculate** the number of training samples
numbatches = m / opts.batchsize;
2: **IF** rem(numbatches, 1) \neq 0
error('numbatches not integer');
End //rem: Remainder after division, rem(x, y) is $x - n * y$

$$\text{Loss} \leftarrow \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^n \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^n \beta_j^2$$

3: Initialize net.rL net.rL = [];
4: **FOR EACH** i = 1 : 1 : opts.numepochs
5: **DO** Model Selection kk = randperm(m); // $x_j = (x_{j-1} * A + C) \bmod M$
6: **FOR** l = 1 : numbatches // Sign samples and corresponding tags
batch_x = x(:, :, kk((l - 1) * opts.batchsize + 1 : l * opts.batchsize));
batch_y = y(:, :, kk((l - 1) * opts.batchsize + 1 : l * opts.batchsize));
// $\frac{\partial \delta}{\partial a_i} = \sum_i \sum_n \frac{\partial \delta}{\partial f(y_i)} \frac{\partial f(y_i)}{\partial a_i}$
net = CNNNET(net, batch_x); // Calculate net output in CNNNET.
net = CNNTRAIN(net, batch_y); // Train algorithms.
net = CNNAPPLYGRADS(net, opts); // Update weights of networks.
7: **END** net.rL(end + 1) = 0.99 * net.rL(end) + 0.01 * net.L; // Record the error.
8: **END** **Return** net.layers

III Solution CNN

III.1 Détection de signalisation



Dans la figure ci-dessus la classification des objets par classes ou groupes c'est la partie à effectuer en premier dans chaque CNN, une détection de trafic dans le but de lire les panneaux de signalisation.

En effet on doit être au courant de tous les événements dans un champ de vision pour savoir comment se comporter dans notre cas c'est les objets sur la route(voitures,camion) et les panneaux ...etc

En règle générale, les neurones sont organisés en couches et les connexions sont établie entre des neurones provenant uniquement de couches adjacentes. Le vecteur d'entité de bas niveau d'entrée est placé dans la première couche et passer d'un calque à l'autre, est transformé en vecteur d'entités de haut niveau.

La quantité de neurones de la couche de sortie est égal au nombre de classes de classification.

Ainsi, le vecteur de sortie est le vecteur de probabilités montrant la possibilité que le vecteur d'entrée appartienne à une classe correspondante.

Un neurone artificiel implémente l'additionneur pondéré, dont la sortie est décrite comme suit:

$$a_i^j = \sigma(\sum_k a_k^{i-1} w_k^i j) \quad (1)$$

Le processus de formation consiste à minimiser la fonction de coût avec des méthodes de minimisation basées sur le gradient décent également connu sous le nom de rétropropagation.

Dans les problèmes de classification, la fonction de coût la plus utilisée est la croix entropie:

$$H(p, q) = -\sum_i Y(i) \log y(i) \quad (2)$$

La formation de réseaux avec un grand nombre de couches, également appelés réseaux profonds, avec activation sigmoïde est difficile en raison à disparaître problème de gradient.

Pour surmonter ce problème, la fonction ELU est utilisée comme fonction d'activation :

$$ELU(x) = \begin{cases} \exp(x) - 1, & x \leq 0 \\ x, & x > 0 \end{cases} \quad (3)$$

Pour résoudre la tâche de reconnaissance des panneaux de signalisation, nous avons utilisé la bibliothèque d'apprentissage en profondeur TensorFlow. Formation et tests ont été mis en œuvre en utilisant l'ensemble de données, La méthode développée permet de classer les 16 plus populaires types de panneaux de signalisation.

L'algorithme développé dans l'article a également été testé sur les images vidéo obtenues dans les rues à l'aide d'un appareil Android Tablette Nvidia Shield intégrée à une voiture.

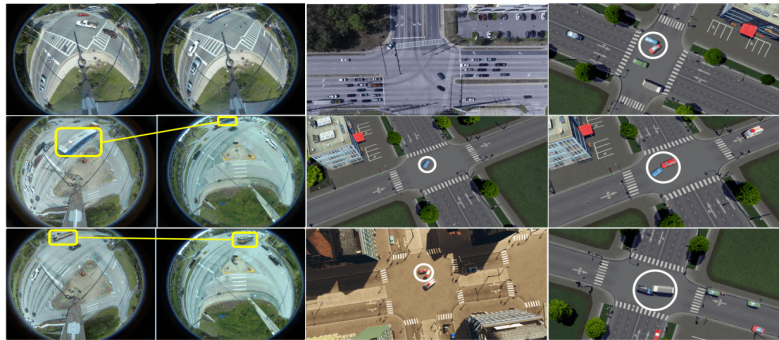


La figure montre les fragments des images originales avec des panneaux de signalisation marqués sur l'image.

III.2 Détection de collision

La reconnaissance des événements dans les scènes naturelles pose un défi pour les approches d'apprentissage en profondeur de la reconnaissance des activités, car un nombre insuffisant d'exemples de formation sont généralement disponibles pour apprendre à se généraliser à toutes les conditions d'observation requises et aux variations d'apparence.

Par exemple, dans les scénarios de conduite, les événements critiques sont souvent fonction de la relation spatiale des objets proéminents, mais les données d'apprentissage sur les événements disponibles peuvent ne pas présenter de variation sur une gamme d'aspects suffisamment large. Il est donc important de développer des modèles de reconnaissance d'activité qui peuvent généraliser efficacement à travers l'apparence des objets et les interactions inter-objets.



La figure montre la détection d'une collision dans un trafic.

Dans cette article l'architecture proposé de réseau convolutionnel à deux flux qui effectue la détection, le suivi et la détection des accidents proches des usagers de la route dans les données vidéo de circulation.

Les réseaux convolutionnels à deux flux consistent en un réseau de flux pour détecter les véhicules individuels et probablement près des régions accidentées au niveau de la trame unique, en capturant fonctionnalités d'apparence avec une méthode de détection d'objets de pointe.

III.3 Détection de dommages matériel

La détection automatisée des dommages extérieurs aux voitures et leur quantification (gravité des dommages) aideraient les concessionnaires de voitures d'occasion (Marketplace) à évaluer les voitures avec précision et rapidité en éliminant le processus manuel d'évaluation des dommages.

Le concept est tout aussi avantageux pour les assureurs, en termes de règlement plus rapide des sinistres et donc de plus grande satisfaction client. Dans cette partie on va parler de la détection de rayures de voiture à l'aide de l'apprentissage par transfert CNN tirant parti du backend Tensorflow.

- Extraction des régions d'intérêt (ROI): l'image est transmise à un réseau CNN qui renvoie la région d'intérêts basée sur des méthodes telles que la recherche sélective (RCNN) ou RPN ..
- les régions sont transmises à un réseau entièrement connecté qui les classe en différentes classes d'images.
Dans notre cas, il s'agira de rayures («dommages») ou d'arrière-plan (carrosserie sans dommage).
- Enfin, une régression de boîte englobante (BB) est utilisée pour prédire les boîtes englobantes pour chaque région identifiée pour resserrer les boîtes englobantes (obtenir BB exact définissant les coordonnées relatives)

Le masque R-CNN est un modèle de segmentation d'instance qui permet d'identifier la délimitation pixel par pixel pour la classe d'objets qui nous intéresse.

Ainsi, Mask R-CNN a deux grandes tâches: 1) détection d'objet basée sur BB (également appelée tâche de localisation) et 2) segmentation sémantique, qui permet de segmenter des objets individuels au pixel d'une scène, quelles que soient les formes.

Ensemble, ces deux tâches Masque R-CNN obtient la segmentation d'instance pour une image donnée.

III.4 Voiture autonome

La figure montre un schéma de principe simplifié du système de collecte des données d'apprentissage de DAVE-2.

Trois caméras sont montées derrière le pare-brise de la voiture d'acquisition

de données, et la vidéo horodatée des caméras est capturée simultanément avec l'angle de braquage appliqué par le conducteur humain.

La commande de direction est obtenue en tapant sur le bus CAN (Controller Area Network) du véhicule.

Afin de rendre notre système indépendant de la géométrie de la voiture, nous représentons la commande de direction comme $1 / r$, où r est le rayon de braquage en mètres.

Nous utilisons $1 / r$ au lieu de r pour éviter une singularité lors de la conduite en ligne droite (le rayon de braquage pour la conduite en ligne droite est l'infini). $1 / r$ passe en douceur par le biais de zéro des virages à gauche (valeurs négatives) aux virages à droite (valeurs positives).

IV Mon avis

Pour résoudre notre problème d'accident de la route il sera indispensable de se dérouter vers des voitures autonome puisque les recherches actuelles font que prouver la stabilité de ces algorithmes.

Au final des points qu'on gagne à travers l'automatisation des voitures mise à part la réduction des accidents et d'embouteillage c'est on a plus besoin de permis de conduire pour conduire.

V Conclusion

Les travaux faits dans les articles sont vraiment très intéressante il serait important d'étudier d'autres algorithmes d'apprentissage automatique pour la gravité des accidents de circulation et appliquer ces algorithmes sur différents ensembles de données pour mettre en compétition ces algorithmes sachant que l'avancé de **Tesla** par rapport aux autres marques fera la différence dans le future proche.