Sûreté de Fonctionnement

« Deep Learning : Forces, Faiblesses & Défis »

Master IRVEA

Année 2022/2023

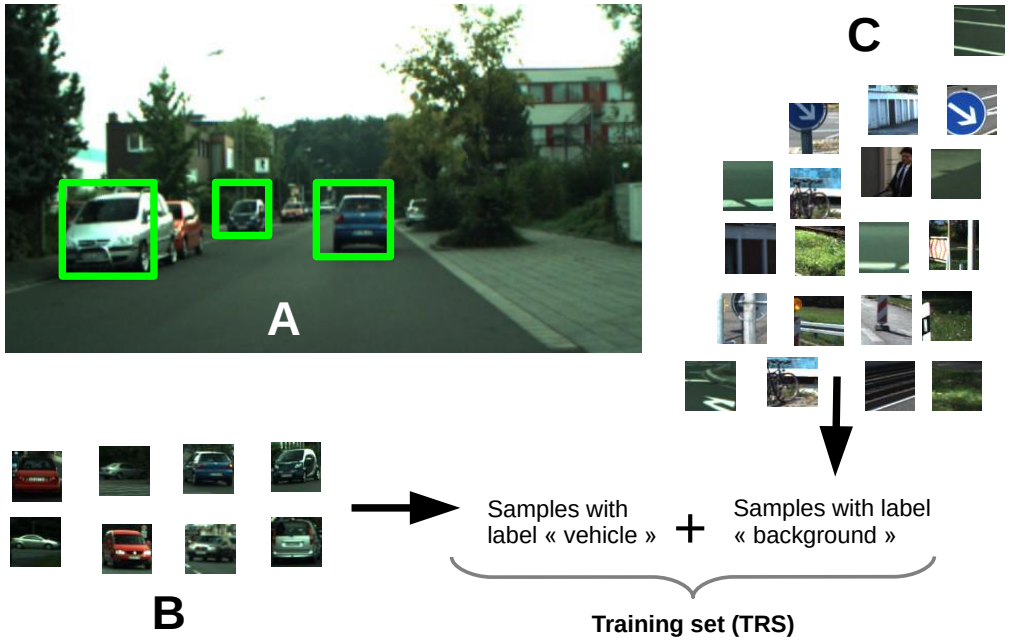
Séance du 10 févier 2023

Présentation de l’Apprentissage & Du « Deep Learning »

Apprentissage automatique

Apprentissage supervisé

- Construire une fonction Y=f(X) à partir d'exemples
- Ex: Classification d'images



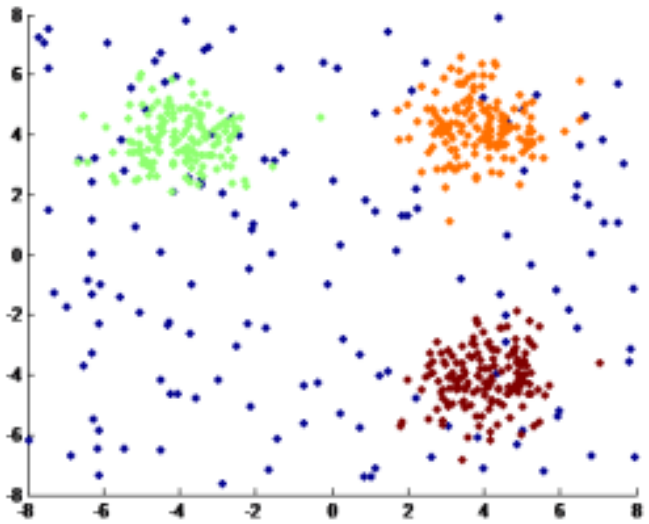
- En général pré-traitement des données -> caractéristiques
- Nombreux algorithmes : SVM, boosting, Réseaux de neurones …

Apprentissage automatique

Autre types d'apprentissage

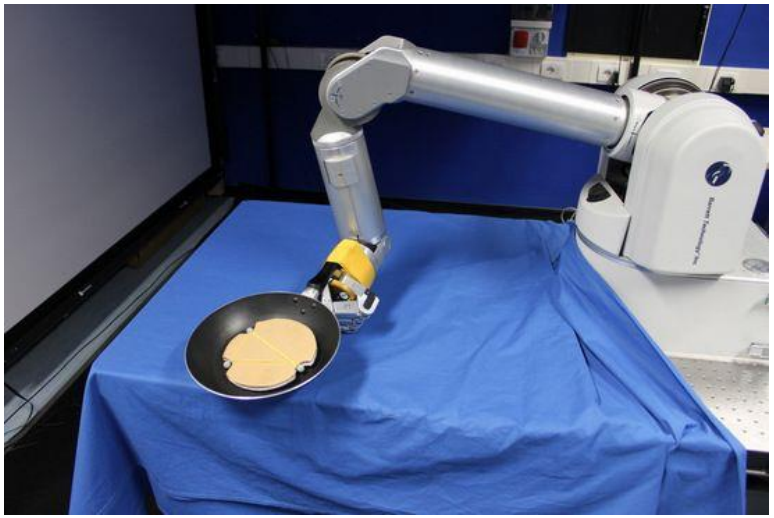
- Non supervisé :

recherche de structure dans les données



- Par renforcement :

recherche de comportement optimisant un cout par essais/erreurs



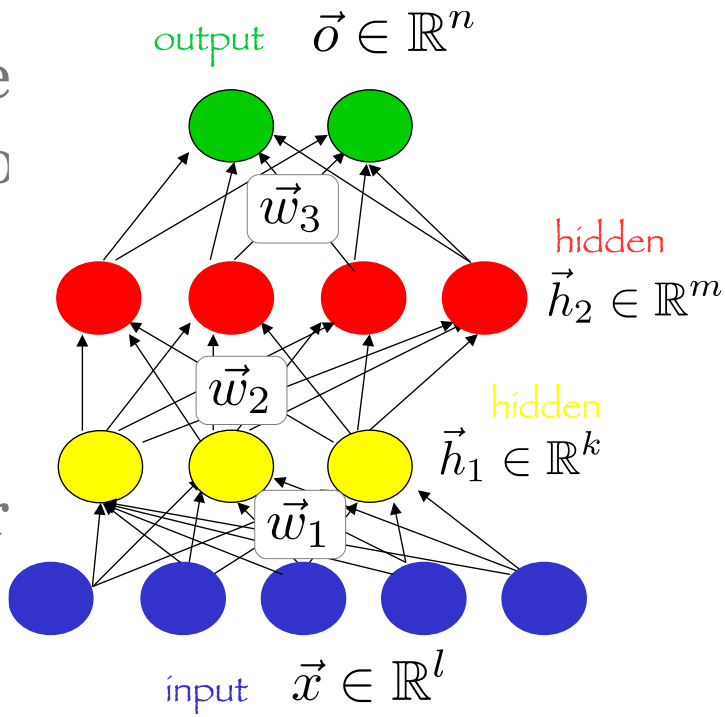
Réseaux de neurones

Famille d'algorithmes (~1960)

- Neurones : unité de calcul élémentaire (somme + non linearité)
- Réseau : neurones connectés par de
- Apprentissage : modification des po
- Méthode : descente de gradient

Différentes structures

- Perceptron multi-couches (Feed for
- Réseaux récurrents
- Extrême learning machines
- ....



Deep Learning

Retour des réseaux de neurones (~2006)

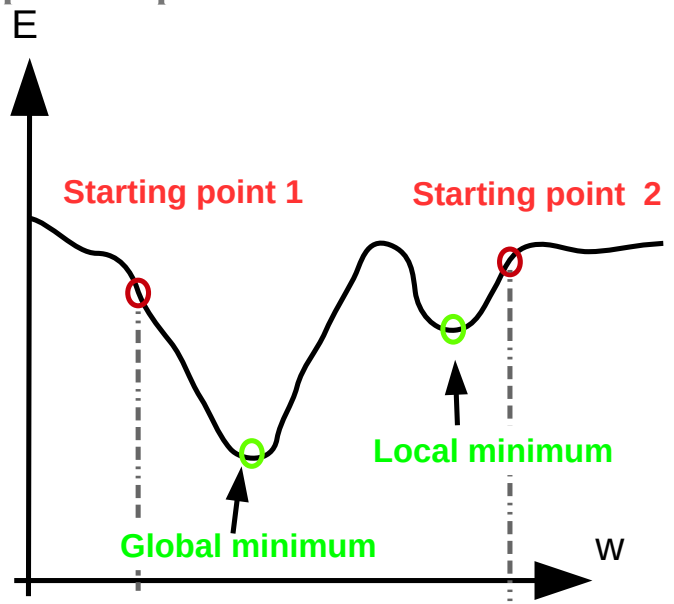
- Perceptron avec « beaucoup » de couches
- Ex: Resnet -> 152 couches
- Base théorique *très* similaire aux perceptrons

Avantages

- Permet des fonctions plus comple
- Etat de l'art sur de nombreux pb.

Problèmes pré 2006

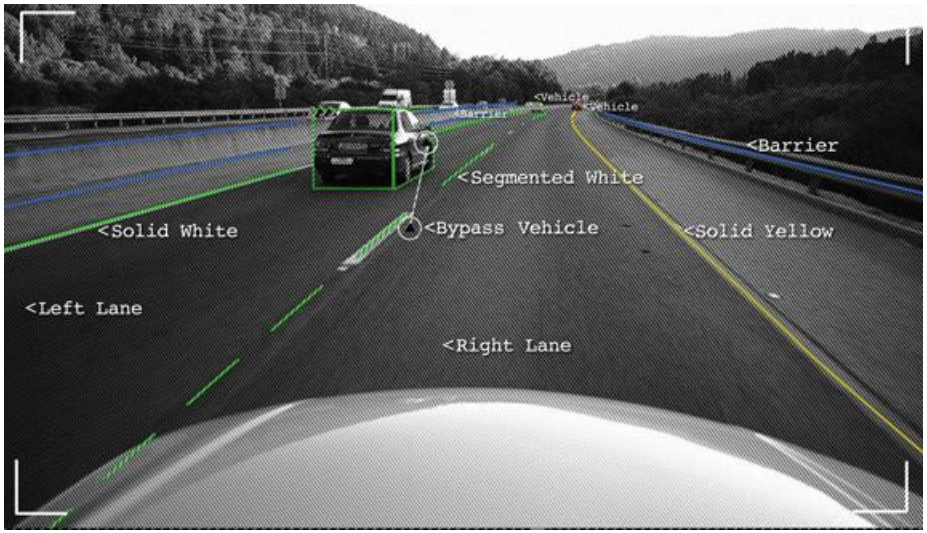
- Puissance de calcul nécessaire
- Apprentissage (min. locaux)
- Données d'apprentissage



Deep Learning : Forces

Excellentes performances applicatives, beaucoup d'applications

- Nombreuses tâches de vision : record en détection, reconnaissance
- Algorithmes de jeux
- Contrôle de robots
- Reconnaissance de la parole
- Traduction automatique
- Description d'images
- …



Deep Learning : Faiblesses

Questions théoriques

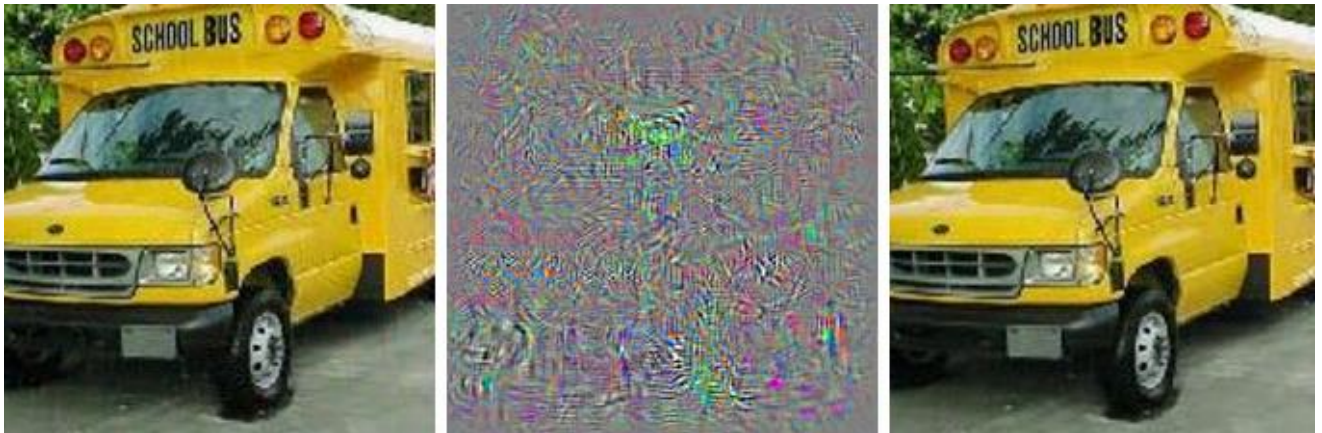
- Choix des modèles largement empiriques
- Hyper-paramètres long et complexes à régler

Questions pratiques

- Pas d'interprétation probabiliste (estimation de la confiance)
- Exemples inquiétants (problème de validation)

Bus

Autruche

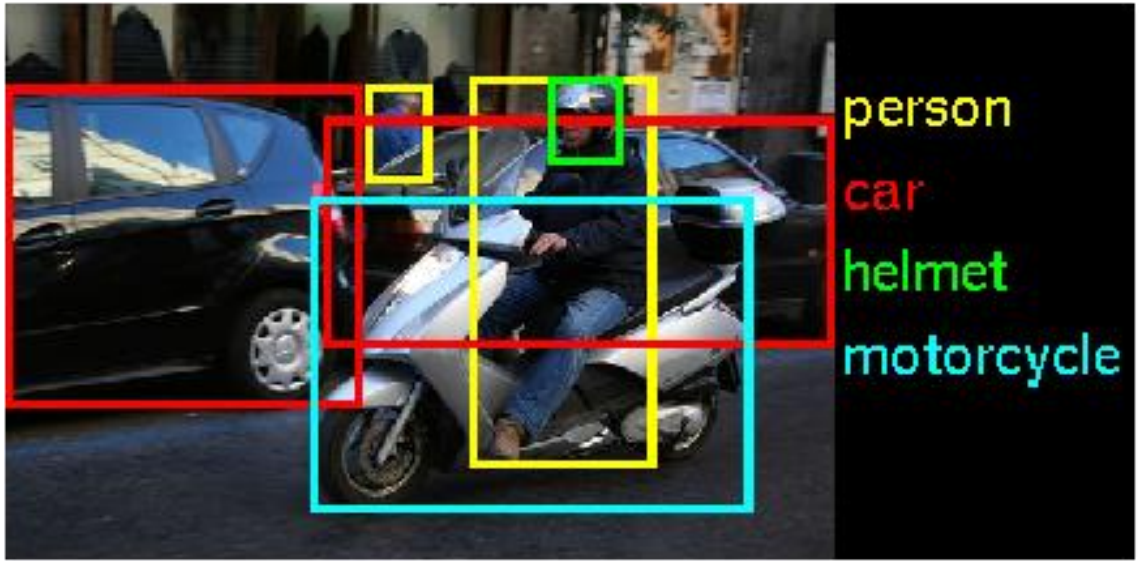


Deep Learning : Faiblesses

Questions pratiques

- Besoin de très grandes quantité de données
  - Réseaux entrainés sur ImageNet (14 million d'images)
  - Mobileye emploie 500 personnes pour annoter des données

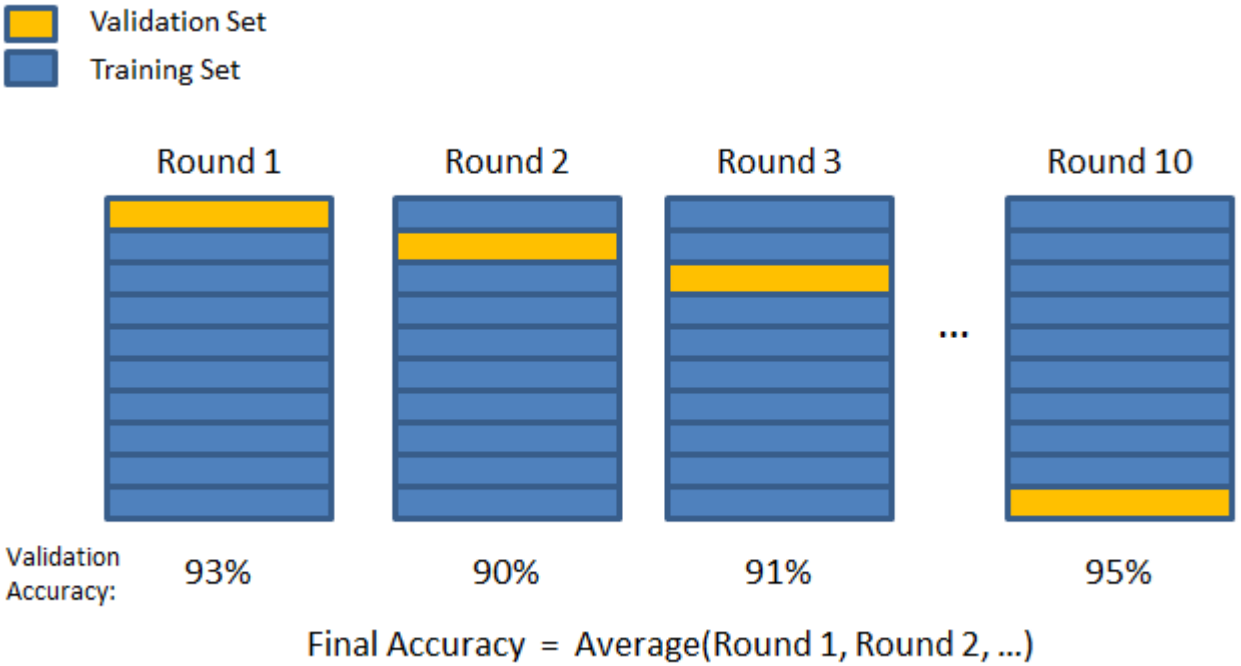
- Puissance de calcul
  - Utilisation extensive de GPU
  - Couteux en embarqué
  - Circuits dédiés



Deep Learning : Applications

Validation des performances

- Pas de possibilité de valider dans l'absolu
- Validation possible sur un jeu de test
- Besoin de beaucoup de données ou cross-validation



Quand ne pas utiliser le deep-learning ?

Problèmes sur des données en petites dimensions

- Une force du deep learning est d'intégrer la détection de caractéristiques pour des données complexes (images, flux audio), ce n'est pas forcement utile
- Exemples :
  - classifications de données statiques simples (télémétrie, gestion de stocks)
  - Problèmes pour lesquels les caractéristiques pertinentes sont connues/simples (détection d'objets connus dans un cadre contraint)
- Alternatives possibles:
  - Séparateurs à Vaste Marge (SVM)
  - Forêt d'arbre aléatoires (Random Forest)
  - Boosting

Quand ne pas utiliser le deep-learning ?

Problèmes pour lesquels l'analyse de la solution est intéressante

- Le deep learning fournit des réseaux de grande taille, souvent difficilement analysables
- La sortie du réseau n'est en général pas interprétable en terme de probabilité
- Pour des problèmes de petites tailles, il peut être utile de comprendre sur quoi se base le résultat et connaître son incertitude
- Alternatives
  - Processus Gaussiens: donnent une solution avec une variance associée
  - Réseaux Bayésien : fournissent un modèle explicite des dépendances entre variables et permettent d'expliquer sur quoi repose le résultat

Quand ne pas utiliser le deep-learning ?

Problème ou la performance n'est pas critique

- Sur certain problèmes, le gain du deep learning peut être réel mais faible au regard de son cout de calcul
- Exemple:
  - Classification de chiffres manuscrits sur la base de données MNIST
  - Meilleurs modèle SVM : 0,56% d'erreur
  - Meilleur modèle Deep Learning : 0,23% d'erreur
  - Gain de 0,33% pour un cout computationnel > 10x
- Alternatives
  - Boosting/cascades
  - Forêt d'arbre aléatoires (Random Forest)

Problematiques d'Implantations et de Validations

Deep Learning : Implémentation

Principe de mise-en-œuvre

- Entrainement du système :

  Détermination des coefficients par exposition à de très grands échantillons
  de données
  de plusieurs millions à plusieurs milliards de paramètres à ajuster.

- Déploiement et calcul :

  Calcul en temps-réel

  Nombre de calculs importants sur des réseaux profonds et des structures complexes.

Possibilité de scinder les plateformes :

Une plateforme d'apprentissage

Une plateforme de reconnaissance

Deep Learning : Implémentation

Spécificité du Deep Neural Network

- Recours à des modèles de réseaux de neurones impliquant une même opération sur un ensemble de neurones (ex: CNN)
- Possibilité de paralléliser par des GPU
- Repose sur des multiplications de matrices
- Ne requiert pas systématiquement une précision importante.

DNN et conception modulaire

- Principe de boîte à outils de différents modèles de réseaux de neurones adaptés à certaines classes ce problèmes.
- Principe de modèle de composition entre les différentes couches et sous-réseaux.

Deep Learning : Implémentation

Utilisation de Hardware Spécifique

- GPU : accélération notable de l'ensemble des opérations matricielles, convolution…
- DSP : accélération des calculs, modèle VLIW.
- FPGA : synthèse de réseau au niveau du composant ou accélérateur de calcul lors de l'apprentissage.
- TPU (Tensor Processor Unit) : ASIC ad hoc pour augmenter le débit des opérations
  - Réduction de la précision de calcul
  - Optimisation du flot d'échange de données
  - Minimisation du contrôle
  - Hybride SIMD et VLIW

Deep Learning : Implémentation

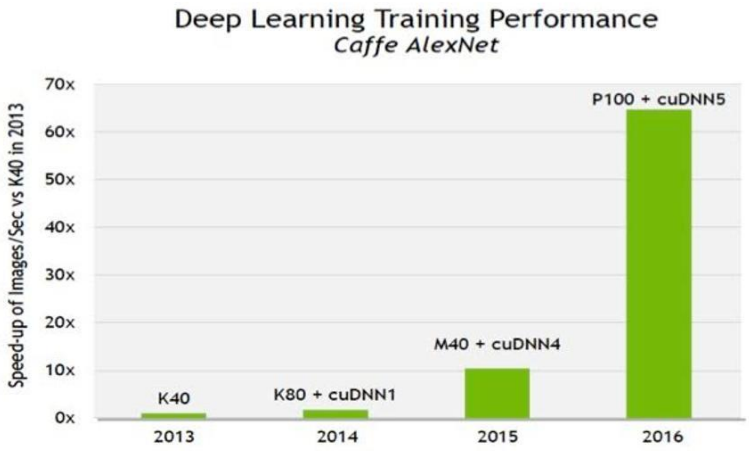
Forces & Faiblesse des briques hardware

- CPU : Faible ratio Performance/Watt.
- GPU : Ratio Performance/Watt élevé.
  Accélération des phases d'apprentissage et des phases de déploiement.
  Adapté plus à certains types de réseau.
  Speedup : x160 pour 130 W, Tesla K40, DnnWeawer
- FPGA : Ratio Performance/Watt élevé (voir très élevé)
  Configuration différente entre apprentissage et déploiement.
  Possibilité de « synthétise » le réseau sur le composant.
  Plus flexible mais plus complexe à mettre en œuvre.
  Speedup : x45 pour 25 W, Arria 10 DnnWeawer
- TPU : Usage pour l'instant uniquement dans la phase de déploiement. Technologie propriétaire adaptée à un Framework.

Deep Learning : Implémentation

Tendances actuelles (au niveau matériel)

- GPU : évolution des GPU pour prendre en compte les DNN et les DCN.
  Augmentation des débits et du nombre de Teraflops
  Accélération d'un facteur 50
  dans les 3 dernières années !
  Diminution des latences
  et Augmentation du Débit
  Enveloppe énergétique
  reste élevée.
- Démonstrabilité du comportement Complexe



AlexNet training throughput based on 20 iterations,
CPU: 1x E5-2680v3 12 Core 2.5GHz. 128GB System Memory, Ubuntu 14.04
M40 bar: 8x M40 GPUs in a node
P100: 8x P100 NVLink-enabled

Deep Learning : Implémentation

- Tendances actuelles (au niveau matériel)
  - CPU : Architecture ManyCore.
    Augmentation de l'efficacité énergétique
    Amélioration des Interconnexions
    Difficilement Qualifiable en terr
    de SdF

  - FPGA : intégration Processeur + FPGA pour SOC
    DSP sur FPGA
    Développement d'outils de synthèses adaptés au DNN et DCN.
    Enveloppe énergétique modérée.
    Problématique de qualification de l'implantation sur FPGA
    Utilisation de Cœurs Démontrables & Temps-réels

Deep Learning : Développement

Emergence de nombreuses solutions d'implantations

Attention 3 phases dans le développement :

Prototype

Apprentissage

Déploiement

Avec des contraintes, des acteurs et des normes différentes.

Deep Learning : Validation

Problèmes :

- Aucune garantie sur la qualité du résultat.
- Absence de « sûreté intrinsèque »

Problèmes complémentaires :

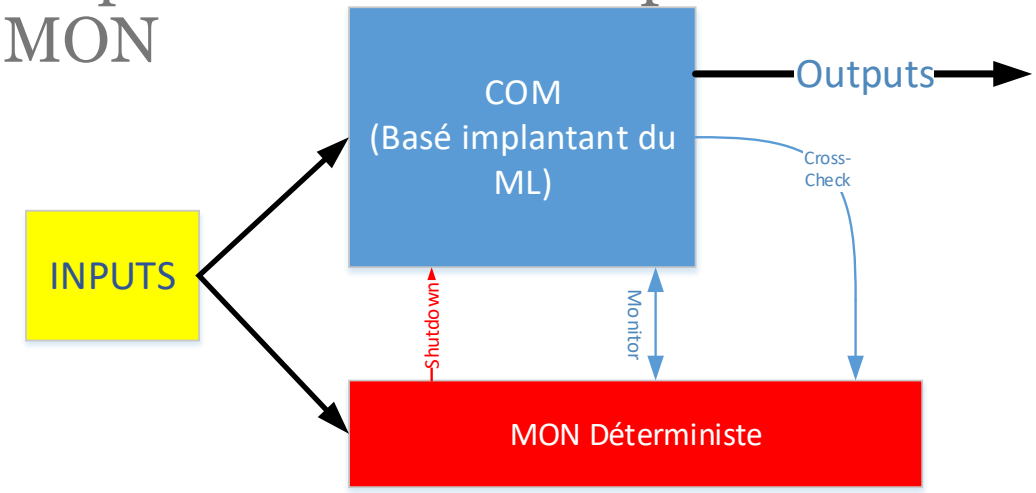
- Apprentissage en ligne : comportement se modifiant dans le temps ? Comment qualifier ?
- Rédaction du code, des bibliothèques (conformité avec Misra C/C++?)

La validation d'un tel algorithme est un problème ouvert à ce jour !

Deep Learning : Validation – Solutions explorées

Architecture de Systèmes basées sur l'autonomie décisionnelle

- Disposer pour chacune des opérations d'une approche COM-MON



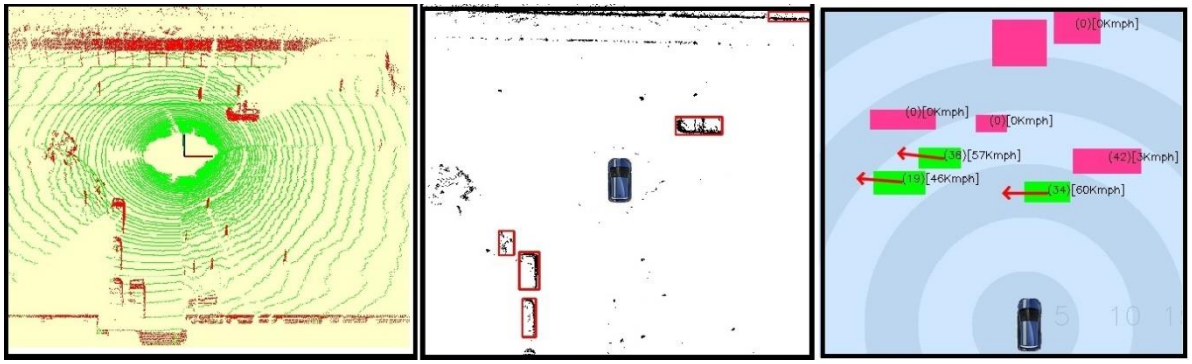
Défis

- Disposer pour les opérations d'un algorithme de vérification de pertinence de la solution.

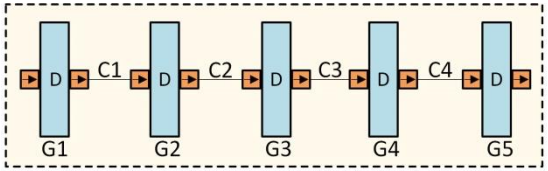
Deep Learning : Validation – Solutions explorées

Nécessité d'une approche Model-Based compositionnelle combinant plusieurs flots de traitement

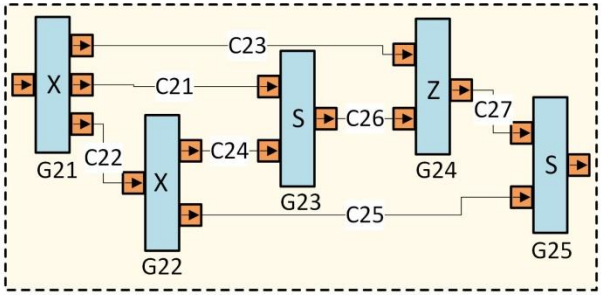
Test Vehicle



Tracking Results



C1:Data Preprocessing, C2: Segmentation, C3: Data Association, C4: State Estimation



C21: 2.5D Projection, C22: Ground Classifier, C24: 2D Projection, C25: Euclidean Clustering, C26:Image Segmentation, C27: Box Mapping

Conclusions

- Le « Machine Learning » ouvre des opportunités nouvelles
  - Efficacité
  - Qualité des solutions
- Le « Machine Learning » pose des problèmes en termes de validation
  - Niveau Logiciel
    - Algorithmes non-déterministes
    - Aucune garantie sur la pertinence des solutions
    - Impossible de qualifier le code
  - Niveau matériel
    - Briques de calcul complexes fortement parallèles
- L'implantation du « Machine Learning » impose de repenser la validation
  - Au niveau Système
  - Au niveau « Exigences de Vérifications »

« SYSTEMES AUTONOMES, VALIDATION & QUALIFICATION»

Master IRVEA
Année 2021/2022
Séance du 7 avril 2022

Différences entre Systèmes Autonomes & Systèmes Automatisés (1)

- Un système automatisé
  - Connaissance a priori de son environnement
  - Connaissance des règles de prise de décision
    - Si vecteur X en entrée, vecteur Y pour action en sortie
  - Reproductibilité de la décision

Différences entre Systèmes Autonomes & Systèmes Automatisés (2)

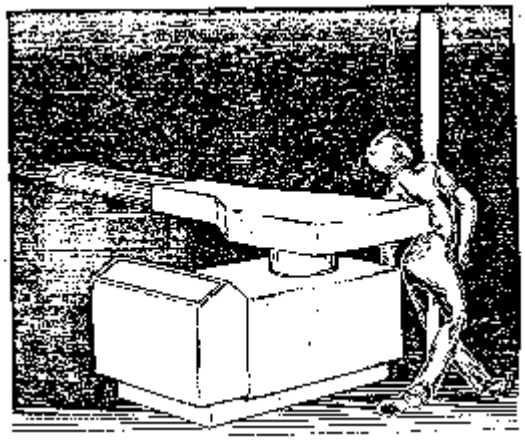
- Un Système Autonome
  - Absence de connaissance a priori de l'environnement
  - Règles de prise de décision
    - Soit déterministe, pour une situation donnée, génération de la même décision
    - Soit non-déterministe, pour une situation donnée, génération de plusieurs décisions possibles
  - Problème de l'explicabilité de la décision
    - Explicable : il est possible de comprendre le calcul ayant conduit à la décision
    - Non-Explicable : la n'est pas possible d'inférer le pourquoi de la décision a posteriori.

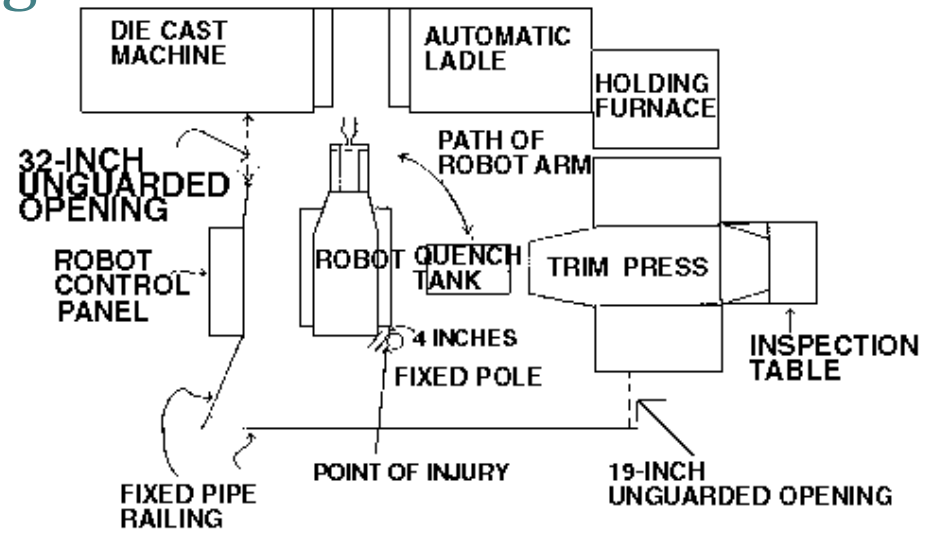
Limites des systèmes automatisés

- Environnement prédéfini
  - Ontologie permettant de caractériser les informations à posséder et à réaliser.
  - Ensemble de capteurs/estimateurs nécessaires à la détermination des informations pertinentes
  - Ensemble des actionneurs nécessaires à la réalisation des opérations nécessaires.
- Décision
  - Identification du domaine de validité du moteur de décision
  - Capacité de détecter les sorties de domaine => mise en sécurité du système.

Origine des fautes dans les Systèmes Automatisés (1)

- Violation d'une contrainte de fonctionnement
  - Ex: obstacle sur la voie quand la voie est supposée vide.
  - Ex: non-respect d'une règle d'exploitation.





NIOH FACE 8420 Fatal Accident Summary Report: Die Cast Operator Pinned by Robot

Origine des fautes dans les Systèmes Automatisés (2)

- Flux d'informations erronées
  - Dysfonctionnement d'un capteur, d'un estimateur de mesure.
  - Corruption de données.
  - Injection de fausses données
- Erreur de prise de décision
  - Erreur d'implantation logicielle (Ariane 501)
  - Erreur de la logique de prise de décision

Origine des fautes dans les Systèmes Automatisés (2)

- Défaillance de l'interaction « Homme-Machine »
  - Confusion (Mauvaise Analyse ou mauvaise compréhension des informations)
  - Mauvaise interaction
  - Mauvaise décision

Validation & Vérification des Systèmes Automatisés (1)

- Validation & Vérification du bon fonctionnement
  - Tests sur un ensemble de scénarii représentatifs.
  - Preuves formelles si nécessaires.
- Règles d'exploitation
  - Garantir l'absence de violation du domaine d'exploitation
- Mise en place de systèmes de protections
  - Prévenir les violations des règles d'exploitation.
  - Identifier les violations et mettre en sécurité.
  - Monitorer les prises de décisions.

Validation & Vérification des Systèmes Automatisés (2)

- Améliorer le design en fonction de l'incidentologie
  - Analyse des défaillances, natures
  - Prise en compte dans le système ou ses évolutions
- Bilan:
  - Concevoir un Système Automatisé est maîtrisé mais peut-être très couteux.
  - La concurrence des fonctions automatisés peut conduire à l'émergence de comportements non-désirables qui doivent être identifiés lors de la conception.

Quand parlons nous de Systèmes Autonomes

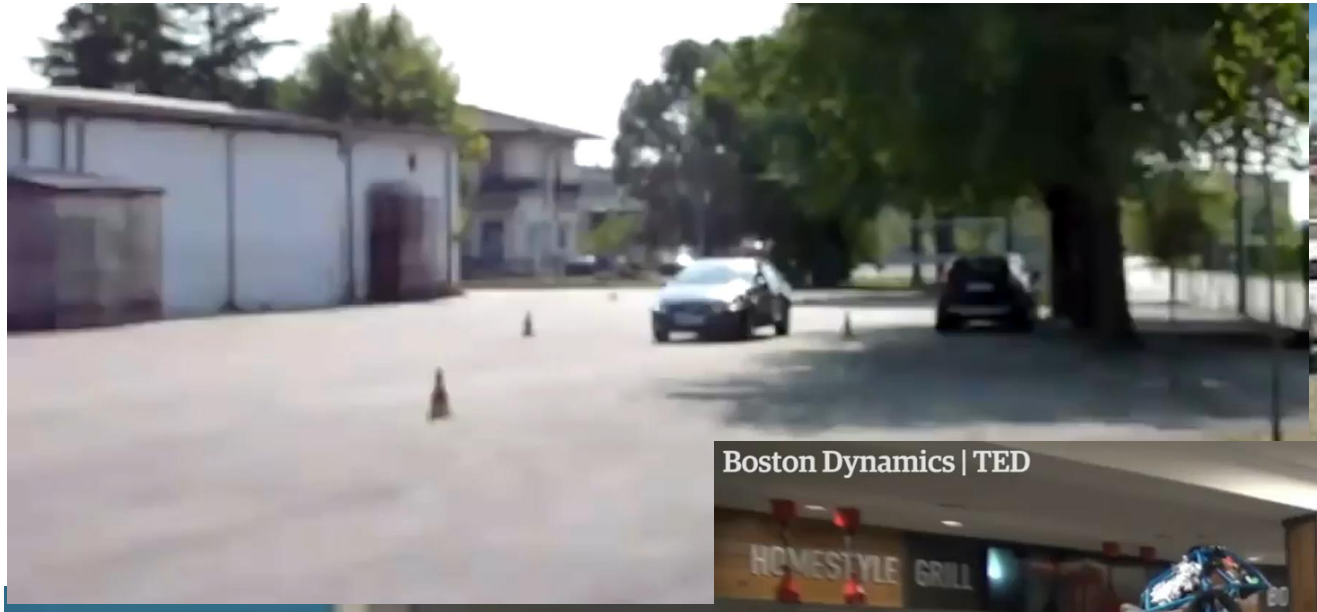
- Augmentation de la capacité à prendre une décision
  - Dans un environnement « nouveau », des conditions « nouvelles ».
  - Dans un environnement complexe (nombres d'interactions importants, acteurs de nature différentes).
- La décision peut-être :
  - Prédéfinie (ie. l'algorithme n'évolue pas dans le temps).
  - Conditionnée (par un apprentissage par exemple).

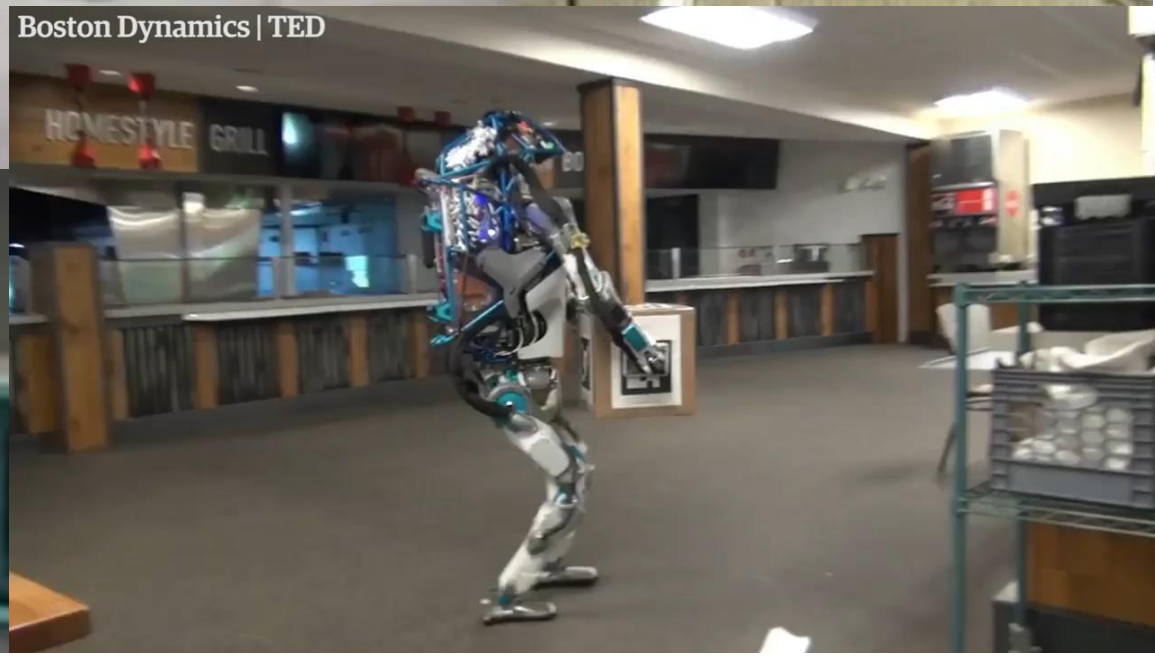
Les Systèmes Autonomes & l'IA

- Systèmes Autonomes n'impliquent pas IA
  - Peut effectuer des traitements complexes sans IO
    - Traitement & analyse d'images
    - Traitement du signal
  - Peut prendre une décision sans IA
    - Système expert (Rule Based)
    - MDP
    - …
  - Intérêt de l'IA pour les systèmes autonomes
    - Richesse et performance de l'analyse de la décision.
    - Simplification de l'implantation !
      - Apprentissage peut-être plus « simple » que développer un algorithme de contrôle-commande complexe de coordination par exemple.

Apprentissage par Renforcement & Génération de contrôleurs

Cependant l’erreur est aussi « Robotique »





Origine des Fautes dans les Systèmes Autonomes

TOUTES LES FAUTES DES SYSTEMES AUTOMATISES

+

TOUTES LES FAUTES LIEES A LA PRISE DE DECISION EN ENVIRONNEMENT INCERTAIN

Problème des la décision dans les Systèmes Autonomes

- Comment caractérise-t-on une bonne décision ?
  - En terme de performances ?
  - En terme d'occurrence de risques ?
  - En terme d'éthique ?
- Comment mesure-t-on la performance d'une décision ?
  - Vitesse d'exécution ?
  - Perception par rapport à ce que ferait l'humain ?
  - Coût/Bénéfice ?

Problème : Optimisation multicritère

- Optimiser la performance
  - Possibilité de rouler sur une route de 2m de large à 200 km/h
  - Problème: Survenance d'une défaillance d'un pneumatique ?
- Optimiser la safety
  - Prendre en compte l'ensemble des évènements quelqu'en soit l'occurrence
  - Problème: comportement « sous-efficace », éventuellement comportement aberrant.
- Optimiser la consomation
- Optimiser la perception sensorielle

Pistes pour définir la qualité d'une décision

- Prendre une « bonne décision humaine » comme référence :
  - Satisfaisante relativement au niveau de prise de risques.
  - Performantes par rapport au critère des « opérateurs qualifiés ».
  - Répondant aux attendes :
    - En terme de confiance,
    - En terme d'appréciation subjective.
  - S'appréciant dans un domaine normatif donné.

Piste pour définir la qualité d'une décision

- Mesurer la différence entre la décision proposée par les système et celle proposée par l'humain
  - Simple dans le cas de jeux
- Plus délicate dans les cas complexes comme :
  - Copilote virtuel
  - Véhicule Autonome

Problématique de la métrique de la performance

- Exemple : Véhicule Autonome
  - Métrique actuelle : nombres d'accidents par millions de km parcourus.
- Limite de la métrique
  - Ne prend pas en compte les défaillances systémiques (a priori sans objet pour l'humain).
  - Ne prend pas en compte la notion de performance de la conduite, son intégration.
  - Ne prend pas en compte la QoS de la conduite autonome.
  - Ne prend pas en compte les variétés de missions réalisées par le véhicule.

Problématique de la mesure de la métrique

- Qualification « a priori »
  - L'estimation de la mesure doit être réalisée a priori.
  - **Problème** :
    - Si la métrique consiste à un nombre de kms parcourus sans accident, comment effectuer ces tests ?
    - Si la métrique consiste en un certain nombre de situations à gérer, comment effectuer ces tests ?

Les problèmes de la décision

- Performances d'un système type reconnaissance de chiffres manuscrits (obtenu par mesure sur un dataset)
  - **Meilleur algorithme** : 0,23% (Deep Learning)
- Problème :
  - Comparaison avec l'humain (à peu près au même niveau de non reconnaissance)
  - L'humain peut prendre une décision supplémentaire
    - Je ne sais pas déchiffrer.
  - Prise en compte de l'évolution de l'écriture dans le temps.
- Performance bien plus mauvaise pour l'analyse de situation plus complexe
  - 80 à 90 % de reconnaissance des objets dans les meilleurs situations.

Les premières réponses (1)

- La validation par les tests réels
  - Mise en condition réelle.
  - Comparer le niveau de performance avec la performance humaine.
  - Estimer le delta objectif et subjectif.
- Limites
  - Uniquement quelques scenarii.
  - Uniquement quelques contextes.

Les premières réponses (2)

- La validation par simulation
  - Réaliser un ensemble de tests dans un simulateur.
  - Comparer le niveau de performance avec la performance humaine.
  - Estimer le delta objectif et subjectif.
- Avantages
  - Réalisation de nombreuses situations.
  - Coût d'évaluation plus modéré.
- Limites
  - Biais induit par l'environnement simulé.
  - Uniquement quelques contextes.

Le biais induit par la simulation

- Différence Environnement simulé/Environnement réel
  - Moins de variabilité.
  - Moins d'information.
  - Structure répétitive.
  - Biais induit sur les capteurs.
- Génération de situations pertinentes/représentatives
  - Simulation de situations dangereuses uniquement -> Difficilement comparable en terme d'accident par km parcourus.
  - Simulation aléatoire de parcours -> problème de représentativité du parcours.

Monde virtuel/Monde réel

Les approches formelles

- **Stabilité de de la réponse :** deux situations relativement proches doivent engendrer la même décision
- Comment garantir une telle stabilité de réponse ?
  - Deep Neural Network : a priori pas de solution lors de la conception.
- Proposition:
  - Introduire un modèle formel approché du modèle de réseaux de neurones employés vérifiant que toute propriété de stabilité dans le modèle formel est correct dans le modèle approché.
  - Démontrer la stabilité du modèle approché
  - En déduire la stabilité du modèle implanté.
- Limite / Défaut
  - Force de l'IA : tenir compte des signaux faibles.

  Risque de masque certains signaux faibles pour conserver la stabilité.

Les approches systèmes

- Panacher plusieurs systèmes de prise de décision [Diversité d'implantation]
- Adjoindre pour chacune des chaînes de prise de décision un estimateur de confiance [Moniteur de QoS]
- Effectuer un voir sur la solution en prenant les différences entre les solutions pondérées par l'estimateur de confiance dans la solution. [Générateur de décision/Détection d'une absence de capacité à répondre]

Les problèmes résiduels

Absence de solutions « satisfaisantes »

« La voiture autonome sera-t-elle programmée pour tuer son conducteur ? »

Pour éviter une collision imminente avec une foule de personnes, quelle décision ?

- **braquer brusquement** au risque de **tuer son conducteur** …
- continuer et **percuter** la foule ?

Conclusions (1)

- Nécessité de développer une quantification de la performances de la décision
  - Définition de nouvelles métriques.
  - Définition de nouvelles méthodes de tests pour estimer cette performance.
  - Comparaison de la métrique humaine vs. la métrique système
- Séparer les deux problématiques de l'implantation et de la décision
  - L'implantation -> Problématique des Systèmes Automatisés
  - La qualification de la décision -> Problématique de la qualité de la décision, validation de celle-ci (à développer)

Conclusions (2)

- Définir la notion de GAME entre Décision d'un Système Autonome et Décision Humaine.
- Développer une nouvelle incidendotologie
  - Distinguer Erreur de Décision/Défaillance de l'implantation.