life.augmented

Présentation IA Embarquée

Qu’est-ce que l’IA ? Pourquoi l’embarquer dans un petit système ?

life.augmented

L’intelligence artificielle (IA) désigne un ensemble de techniques permettant à des machines de reproduire des comportements humains tels que le raisonnement, la planification, la créativité ou la prise de décision automatisée. Embarquer l’IA dans un petit système permet de rendre les appareils autonomes, plus intelligents et capables de traiter des données localement, même lorsque les ressources (mémoire, puissance, énergie) sont limitées.

Intérêt d’embarquer l’IA dans un petit système

L’ajout de l’intelligence artificielle dans de petits systèmes, comme des capteurs ou des objets connectés, permet :

• D’améliorer l’autonomie, car l’IA peut traiter les informations sur le dispositif lui-même, sans dépendre d’un serveur central.

• De réduire la latence : les réactions sont plus rapides, ce qui est utile pour des applications de sécurité, surveillance, ou contrôle en temps réel.

• De consommer moins d’énergie et de bande passante en limitant les échanges de données vers le cloud.

• D’ouvrir de nouvelles applications (ex : reconnaissance vocale en local, analyse de données de santé sur wearable, détection d’objets pour drones).

Nano Edge AI ou STM32Cube AI ?

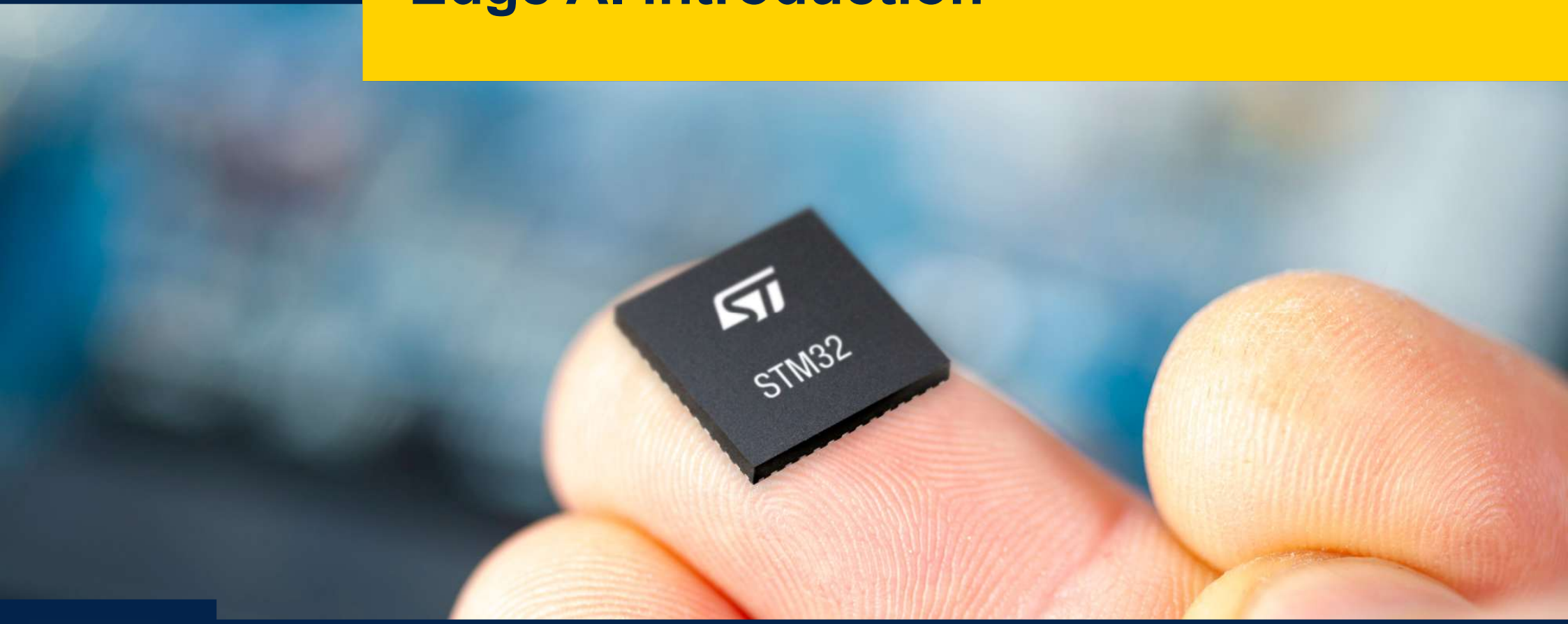
STM32Cube.AI

Objectif: optimiser et compiler des réseaux neuronaux pré-entraînés pour les exécuter sur des microcontrôleurs STM32.
Utilisation: part d'un modèle déjà construit (par ex. PyTorch/TensorFlow) et produit du code C optimisé qui peut être intégré directement dans votre firmware.

NanoEdge AI Studio

Objectif: générer des solutions ML directement à partir de données et sans nécessiter de neural networks pré-entraînés lourds; c'est une démarche AutoML adaptée aux ressources embarquées.
Utilisation: propose une bibliothèque de modèles optimisés et automatisés (détection d'anomalies, classifications, régressions, extrapolation) et peut générer des bibliothèques C prêtes à être utilisées sur les MCU STM32.

Edge AI introduction



life.augmented

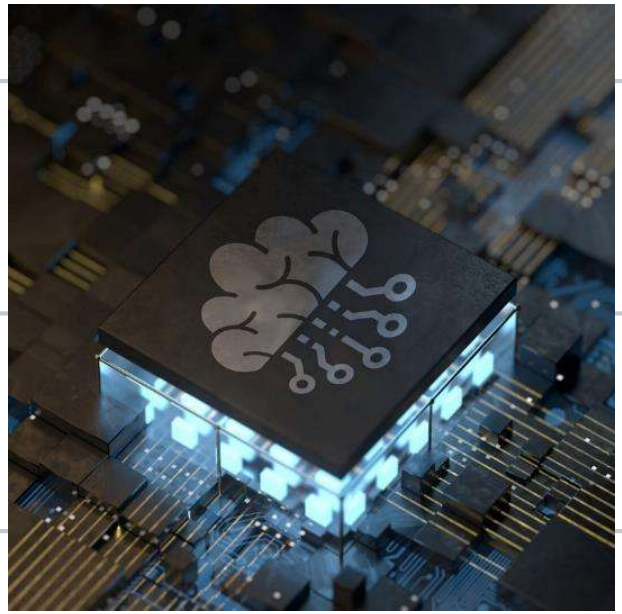
EdgeAI: Benefits

Moving artificial intelligence closer to the signal brings several benefits

Ultra-low latency
Real-time applications

Improved accuracy
Adapt to local environment

Reduced data transmission
Generate meaningful information



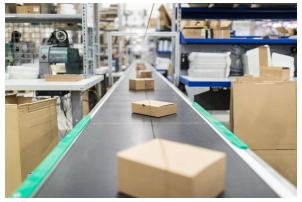
Privacy & security
No raw data shared

Sustainable on energy
Low-power consumption

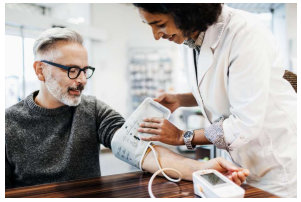
Advanced experience
Personalized features

life.augmented

Edge AI in Different Industries



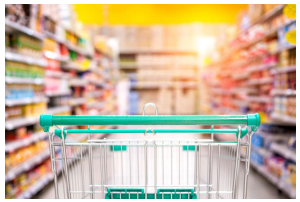
Manufacturing and Industrial Automation



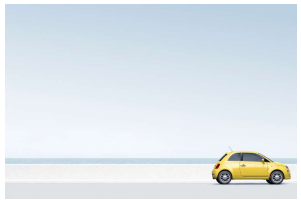
Healthcare

Smart Cities and Infrastructure

Agriculture



Retail



Autonomous Vehicles

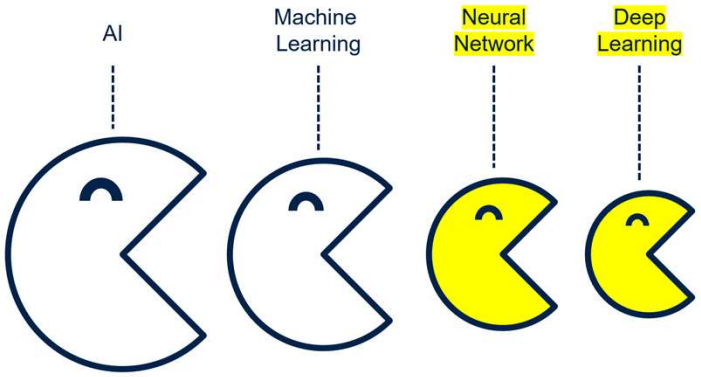
Security and Surveillance

...

Neural Networks

life.augmented

éfinition réseaux neuronaux



Les réseaux de neurones sont composés d'un ensemble de neurones organisés en couches, chacune possédant ses propres poids et biais apprenables.

Les réseaux de neurones servent à reconnaître des motifs dans les données.

Il existe plusieurs types de réseaux de neurones, chacun conçu pour résoudre différents types de problèmes.

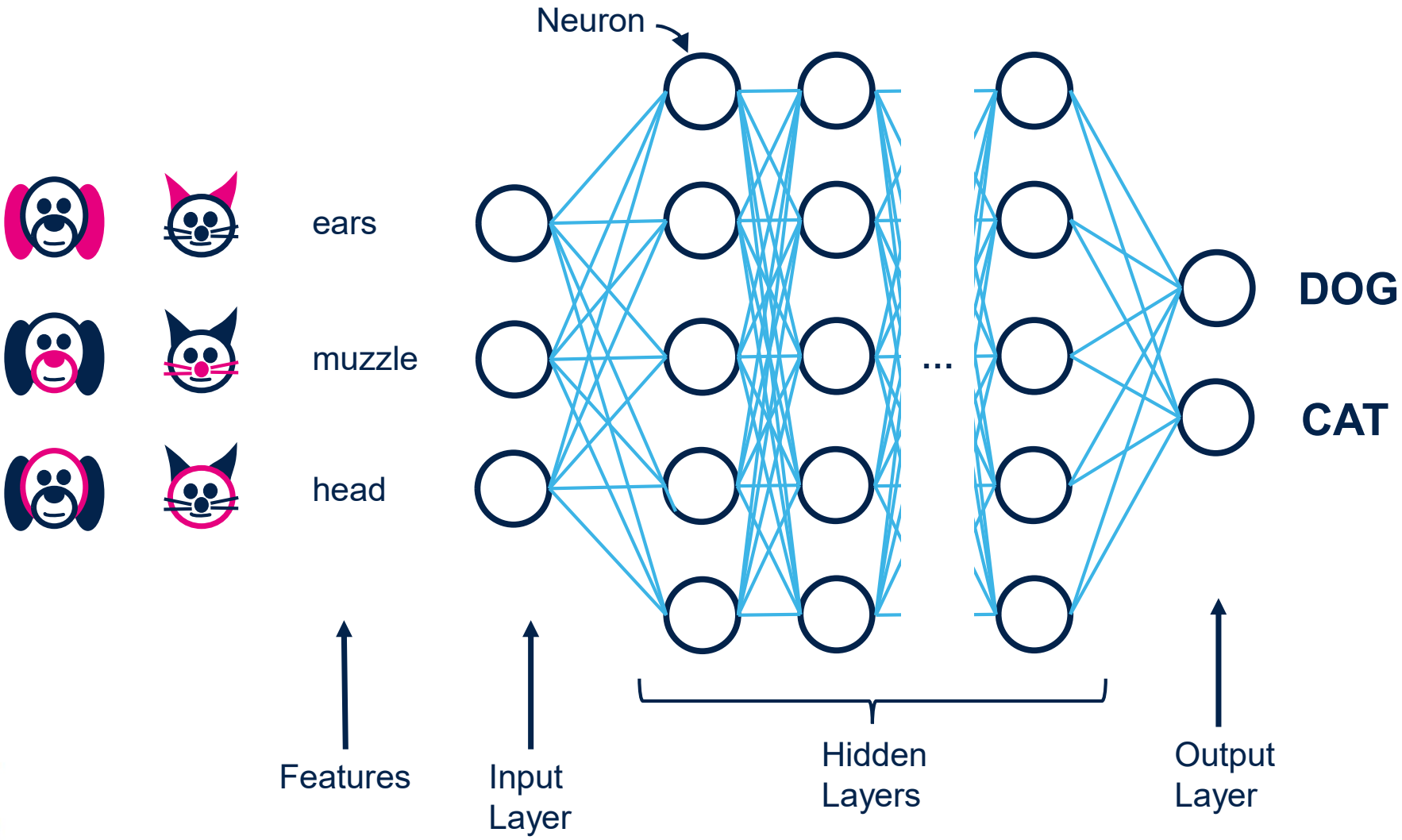
Les réseaux de neurones convolutifs (CNN) sont couramment utilisés pour le traitement d'images et de vidéos.

Les réseaux de neurones récurrents (RNN) sont couramment utilisés pour les données séquentielles, comme celles que l'on rencontre dans le traitement audio ou le traitement automatique du langage naturel (TALN), mais leur entraînement ne peut pas être parallélisé.

Les réseaux de « transformeurs » sont également utilisés pour les données séquentielles ; ils offrent généralement de meilleures performances, mais au prix d'un coût de calcul et d'une consommation de mémoire plus élevés.

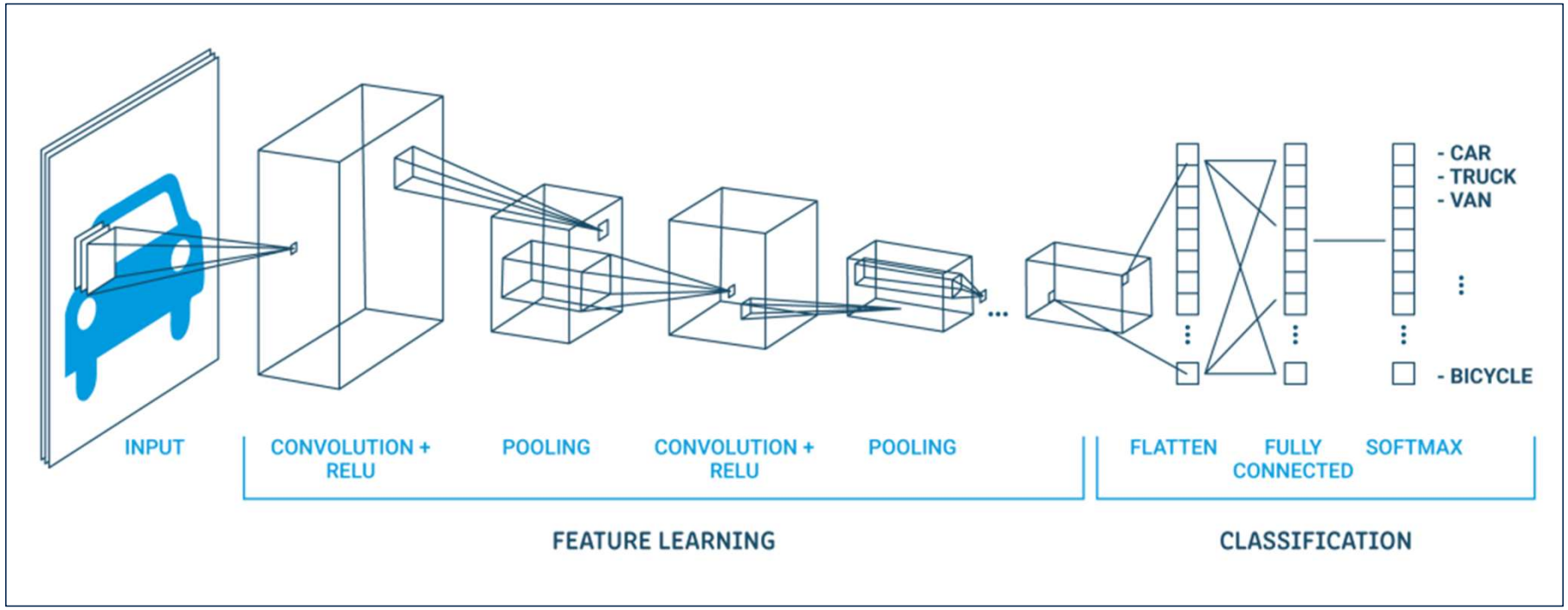
life.augmented

Neural Networks training



life.augmented

Convolutionnal Neural Networks (CNN)

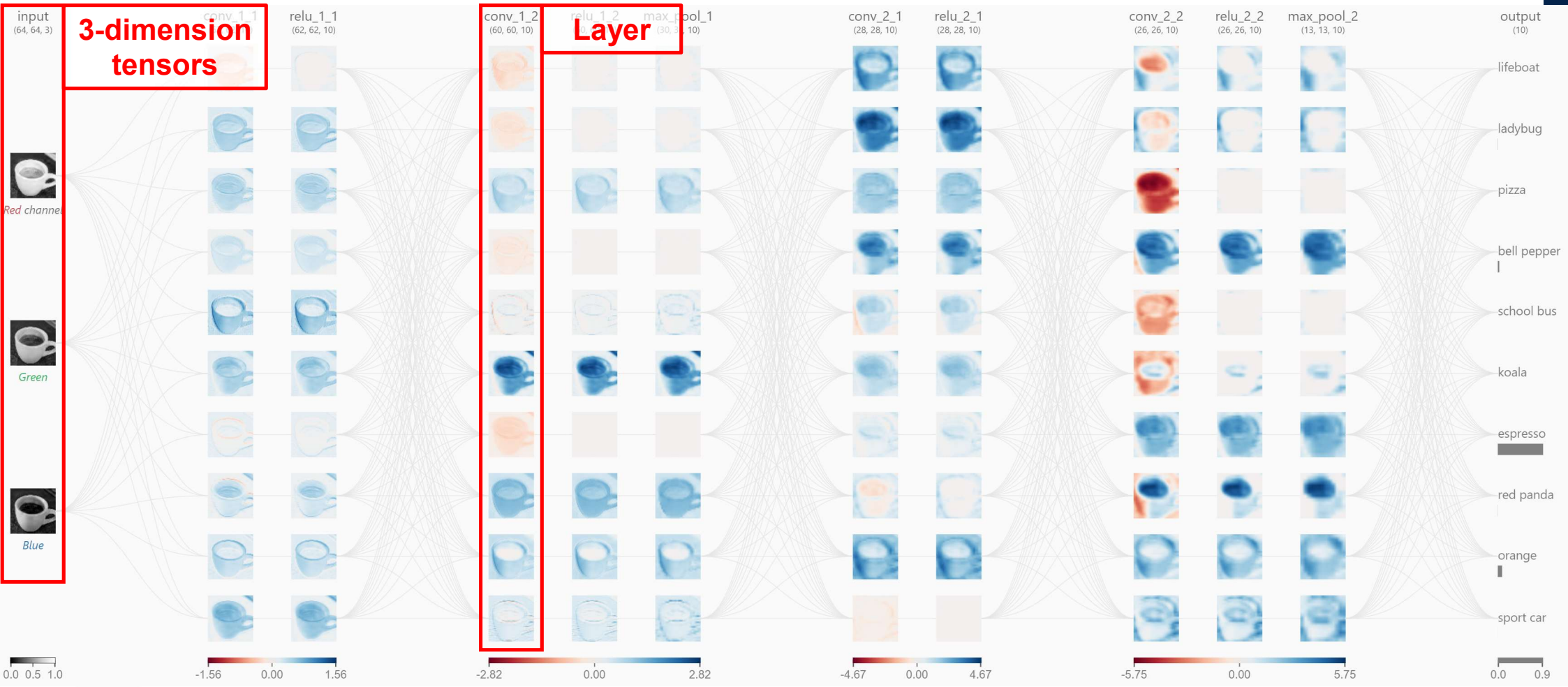


Convolutional Neural Networks (CNN)

On peut décrire un réseau de neurones convolutif (CNN) en trois blocs :

- Tenseurs : matrices n-dimensionnelles.
- Couches : fonctions qui effectuent des opérations différentiables sur les tenseurs. Chaque couche peut prendre une ou plusieurs entrées et produire un ou plusieurs tenseurs en sortie.
- Poids des couches : tenseurs décrivant les paramètres d'une couche. Ils sont ajustés pendant la phase d'entraînement, généralement par descente de gradient, et permettent au modèle de s'adapter au problème et aux données fournies.

Convolutionnal Neural Networks (CNN)



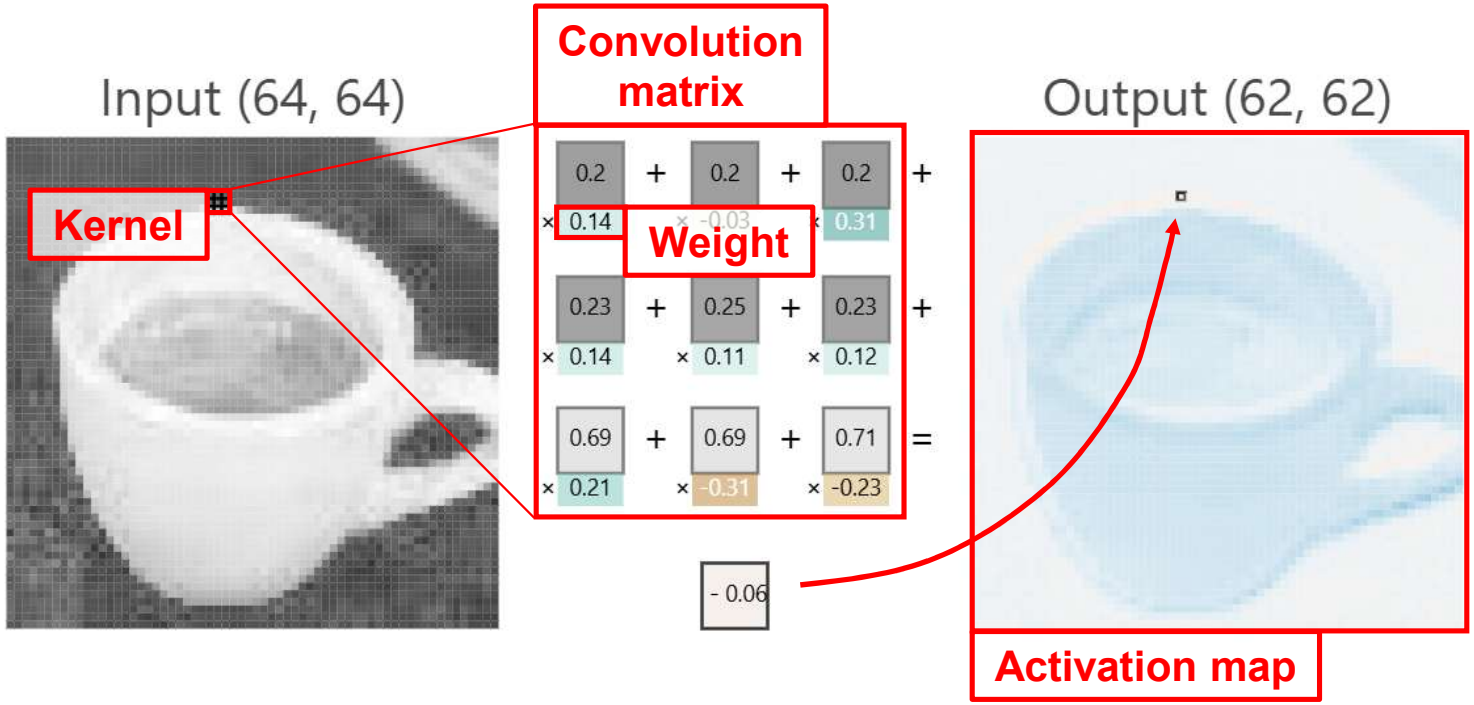
life.augmented

Convolutionnal Neural Networks (CNN)

Convolution layers

Kernel size = 3*3

Les poids sont modifiés à chaque itération.

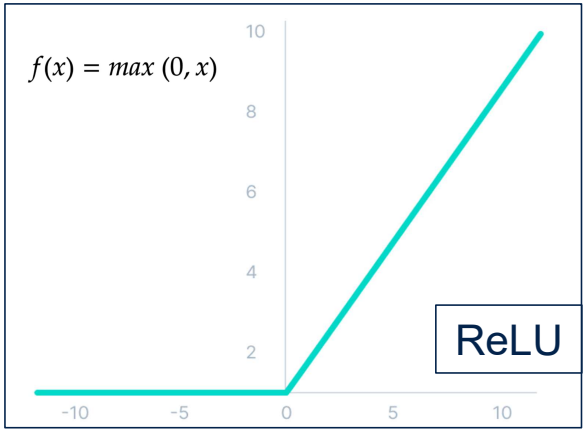


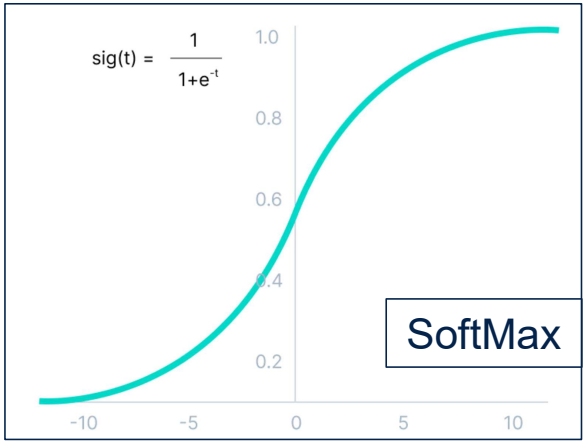
Convolutionnal Neural Networks (CNN)

Fonctions d'activation

- **Transforment la sortie d'une couche en une valeur d'activation à transmettre à la couche suivante.**
- **ReLU : Unité linéaire rectifiée, très efficace en termes de calcul.**
- **SoftMax : Utilisée juste avant la dernière couche, elle renvoie la probabilité de chaque classe (somme des probabilités = 1).**

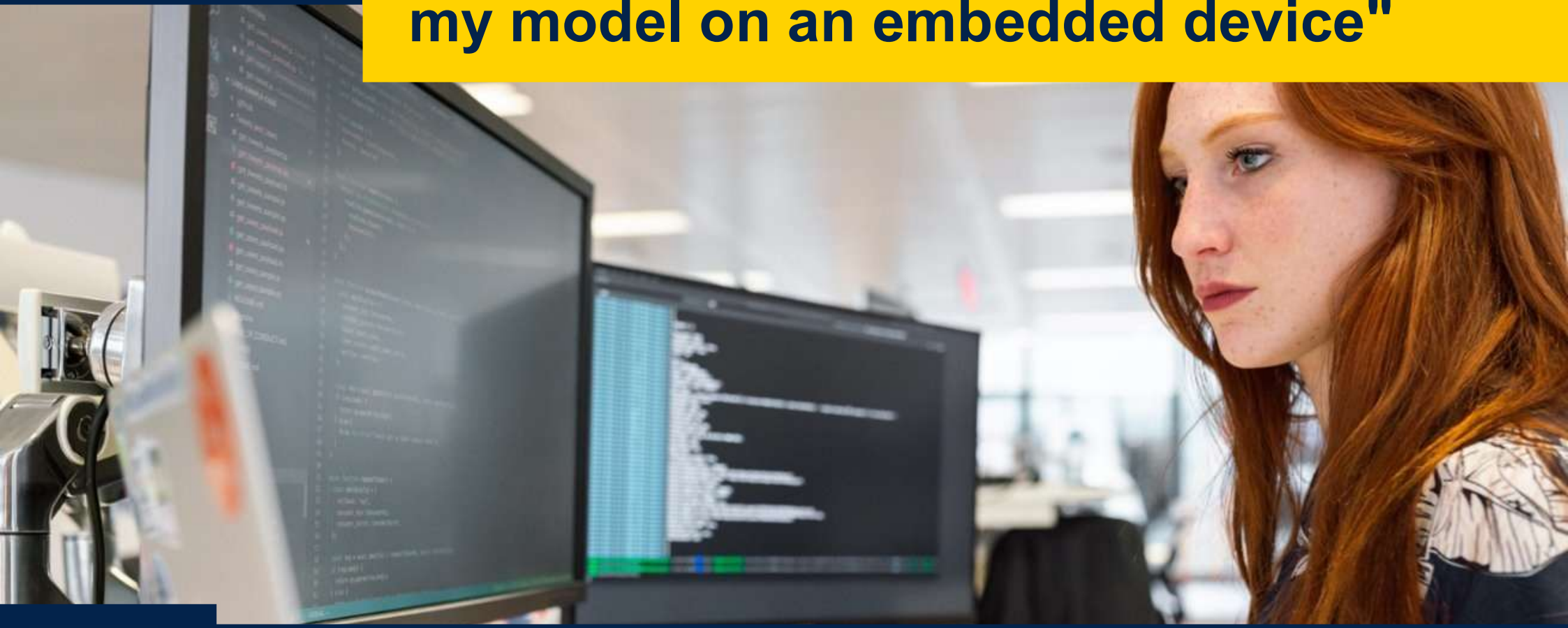
Parmi de nombreuses autres.





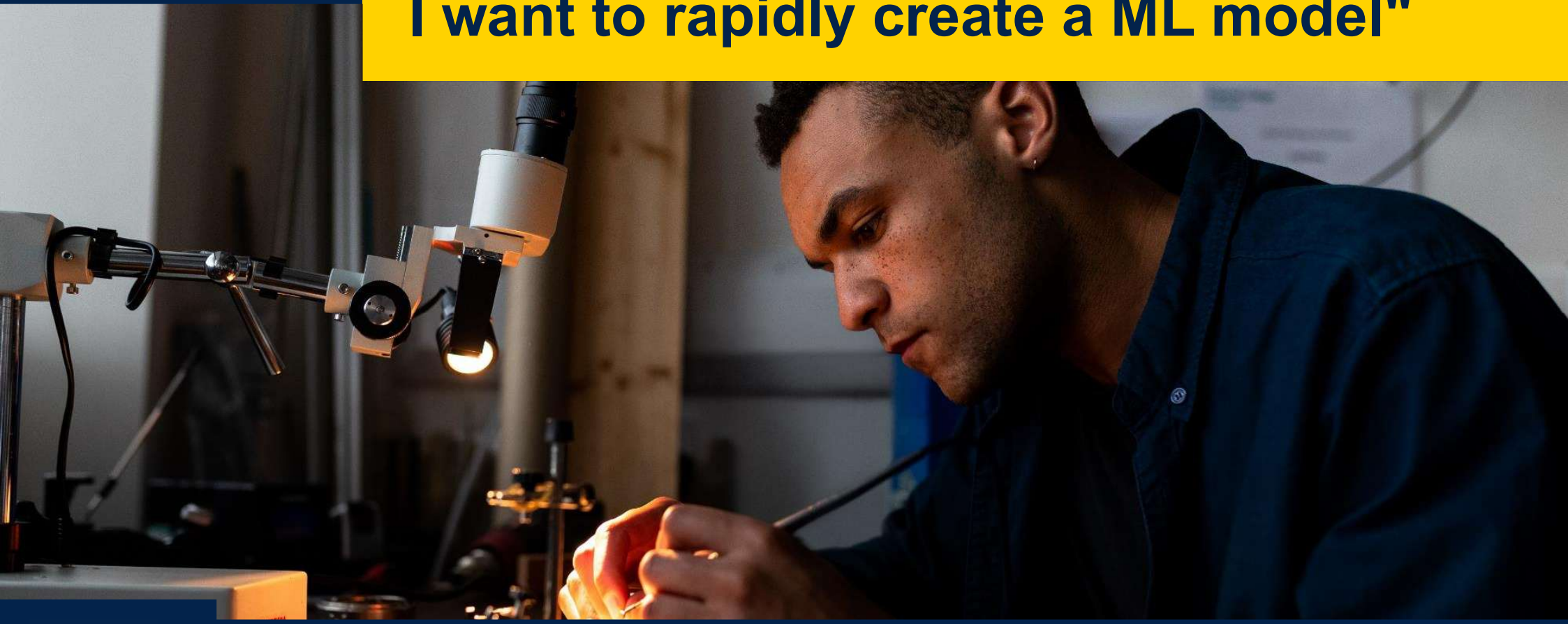
life.augmented

"I am a Data Scientist and I want to run my model on an embedded device"



life.augmented

"I am an embedded developer and I want to rapidly create a ML model"



life.augmented