

# Rapport pour le projet de Deep Learning

## “Classification de Panneaux de Signalisation Routière”

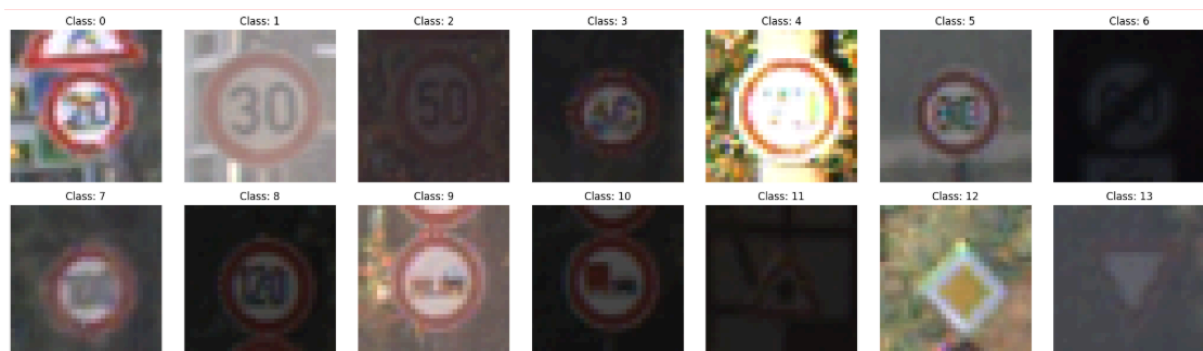
Svetlana TSURANOVA, Luckner JEAN, Aminata FALL, Yaye Touti DIA  
Mars 2025

### 1. Objectif du projet

Dans ce projet, nous avons construit un modèle Convolutional Neural Network pour la classification de panneaux de signalisation routière, qui est extrêmement important dans les systèmes de conduite autonome.

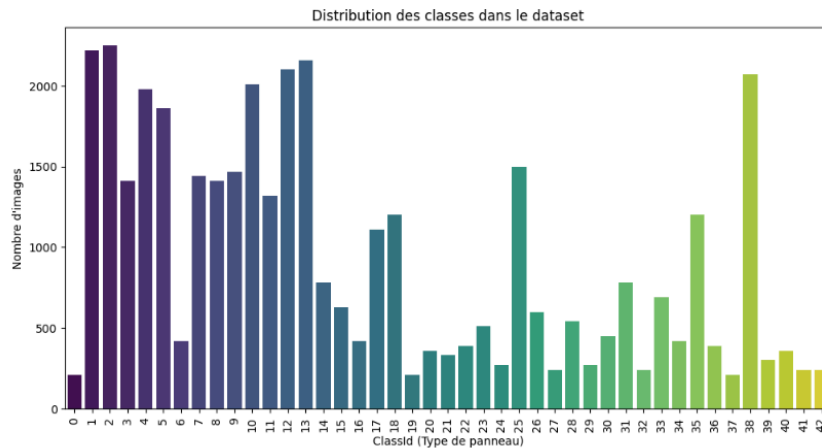
### 2. Exploration et préparation des données

Dataset GTSRB (German Traffic Sign Recognition Benchmark) a été utilisé pour le projet. Il contient 43 classes de panneaux routiers. Voici les exemples des images de dataset:



La répartition des images dans chaque classe est suivante:

Le dataset présente un **déséquilibre des classes**, avec certaines catégories de panneaux largement sur-représentées (ex. classes 1, 2, 11, 13, 38) tandis que d'autres sont sous-représentées (ex. classes 0, 5, 20, 28, 42)



Le dataset n'est pas suffisamment bien documenté, et nous ne pouvons pas dire comment il a été créé. Mais nous voyons que des transformations ont été appliquées à de nombreuses images. Par exemple, 29 images supplémentaires ont été créées à partir d'une seule:



C'est pourquoi, pour le modèle de base, il a été décidé de ne pas utiliser l'augmentation des images.

Lorsque la taille moyenne des images est de  $50,8 \times 50,3$ , nous avons décidé de les redimensionner à  $64 \times 64$  pour ne pas perdre l'information.

La normalisation a été réalisée lors du prétraitement des images. Le dataset a été divisé en ensembles d'entraînement et de validation dans une proportion de 80% pour l'entraînement, 20% pour la validation.

### 3. Conception et Implémentation du Modèle CNN

Différentes architectures de CNN ont été testées: avec 2, 3, 4 couches de convolution, sans et avec Dropout, sans et avec BatchNormalization. Le modèle le plus avancé VGG a été testé. Mais tous les modèles donnent des résultats très proches en termes de l'accuracy, precision, recall, F-score.

Modèle	Couches	Fonctions d'activation	Total params	Dropout	BatchNormalization	val_accuracy	Precision	Recall	F-Score
CNN 01	Conv2D — 2 MaxPooling2D — 2 Flatten — 1 Dense — 2	3 Relu, 1 Softmax	1,650,027	-	-	98,7%	0.978	0.977	0.977
CNN 02	Conv2D — 3 MaxPooling2D — 3 Flatten — 1 Dense — 2	4 Relu, 1 Softmax	329,227	-	-	99,6 %	0.988	0.988	0.988
CNN 03	Conv2D — 3 MaxPooling2D — 3 Dropout — 4 Flatten — 1 Dense — 2	4 Relu, 1 Softmax	376,235	+	-	99,7%	0.995	0.995	0.995
CNN 04	Conv2D — 6 MaxPooling2D — 3 BatchNormalization — 3 Flatten — 1 Dense — 2 Dropout — 1	4 Relu, 1 Softmax	815,051	+	+	98,1%	0.996	0.996	0.996
VGG	Conv2D — 6, MaxPooling2D — 3, Flatten — 1, Dense — 2.	7 Relu, 1 Softmax	1,341,259	-	-	98,9 %	0.99	0.99	0.99

## 4. Évaluation du Modèle

l'architecture de notre modèle comprend:

- `Conv2D(64, (3,3), activation='relu')`
- `MaxPooling2D(2,2)`
- `Dropout(0.2)`
- `Conv2D(64, (3,3), activation='relu')`
- `MaxPooling2D(2,2)`
- `Dropout(0.2)`
- `Conv2D(64, (3,3), activation='relu')`
- `MaxPooling2D(2,2)`
- `Dropout(0.2)`
- `Flatten()`
- `Dense(128, activation='relu')`
- `Dropout(0.3)`
- `Dense(43, activation='softmax')`

L'entraînement a montré une progression de la précision et une diminution de la perte:

- **Précision finale sur validation** : 99.51%
- **Perte finale sur validation** : 0.025

Les courbes d'apprentissage montrent que le modèle atteint une haute précision sans overfitting.

Loss et Accuracy pour le modèle CNN 3 avec l'architecture suivante:

Layer (type)	Output Shape	Param #
conv2d_132 (Conv2D)	(None, 62, 62, 64)	1,792
max_pooling2d_87 (MaxPooling2D)	(None, 31, 31, 64)	0
dropout_27 (Dropout)	(None, 31, 31, 64)	0
conv2d_133 (Conv2D)	(None, 29, 29, 64)	36,928
max_pooling2d_88 (MaxPooling2D)	(None, 14, 14, 64)	0
dropout_28 (Dropout)	(None, 14, 14, 64)	0
conv2d_134 (Conv2D)	(None, 12, 12, 64)	36,928
max_pooling2d_89 (MaxPooling2D)	(None, 6, 6, 64)	0
dropout_29 (Dropout)	(None, 6, 6, 64)	0
flatten_32 (Flatten)	(None, 2304)	0
dense_64 (Dense)	(None, 128)	295,040
dropout_30 (Dropout)	(None, 128)	0
dense_65 (Dense)	(None, 43)	5,547

Total params: 376,235 (1.44 MB)

