Présentation de la lecture d'article

Jules Bataller Beltran Kenewy Diallo Kossi Abotsi
Lorenzo Gaggini

2025-02-10

Sommaire

Contexte

Le cadre statistiques

- Classification supervisée d'images, où chaque image est associée à une étiquette de classe.
- Minimisation d'une fonction de perte par descente de gradient

Contexte

Enjeux de la Visualisation des Réseaux de Neurones Convolutionnels

- Interprétation des décisions du modèle
- Identification des caractéristiques importantes
- Tester la robustesse face aux variations

Principaux outils

Fonctionnement des réseaux de neurones convolutionnels (CNN)

- Les réseaux de neurones convolutionnels (CNN) traitent des données structurées en grilles, comme les images et se compose de plusieurs types de couches :
  - **Couche de convolution** :
    - Applique des filtres sur l'image pour extraire des caractéristiques locales (bords, textures).
  - **Fonction d'activation (ReLU)** :
    - Permet de modéliser des relations complexes.
  - **Couche de pooling** :
    - Rend le réseau plus robuste aux variations (translations, redimensionnements).
  - **Couches entièrement connectées** :
    - Les cartes d'activation sont aplaties et traités comme dans un réseau de neurones classique.

Principaux outils

DeconvNet : Interprétation des CNN

- Le DeconvNet permet d'interpréter les activations d'un CNN en reconstruisant l'image d'origine à partir des caractéristiques extraites.
  - **Unpooling (Dé-pooling)** :
    - Permet de restaurer la structure de l'activation.
  - **Rectification inversée** :
    - Applique l'inverse de la fonction d'activation ReLU utilisée dans le CNN.
  - **Convolution transposée** :
    - Permet de reconstruire une image en visualisant les caractéristiques identifiées par le CNN.
- Cette approche offre une meilleure compréhension des motifs détectés à chaque niveau du réseau, en identifiant quelles parties de l'image d'origine ont influencé les activations.

Résultats expérimentaux

Exemples de visualisations (1/2)

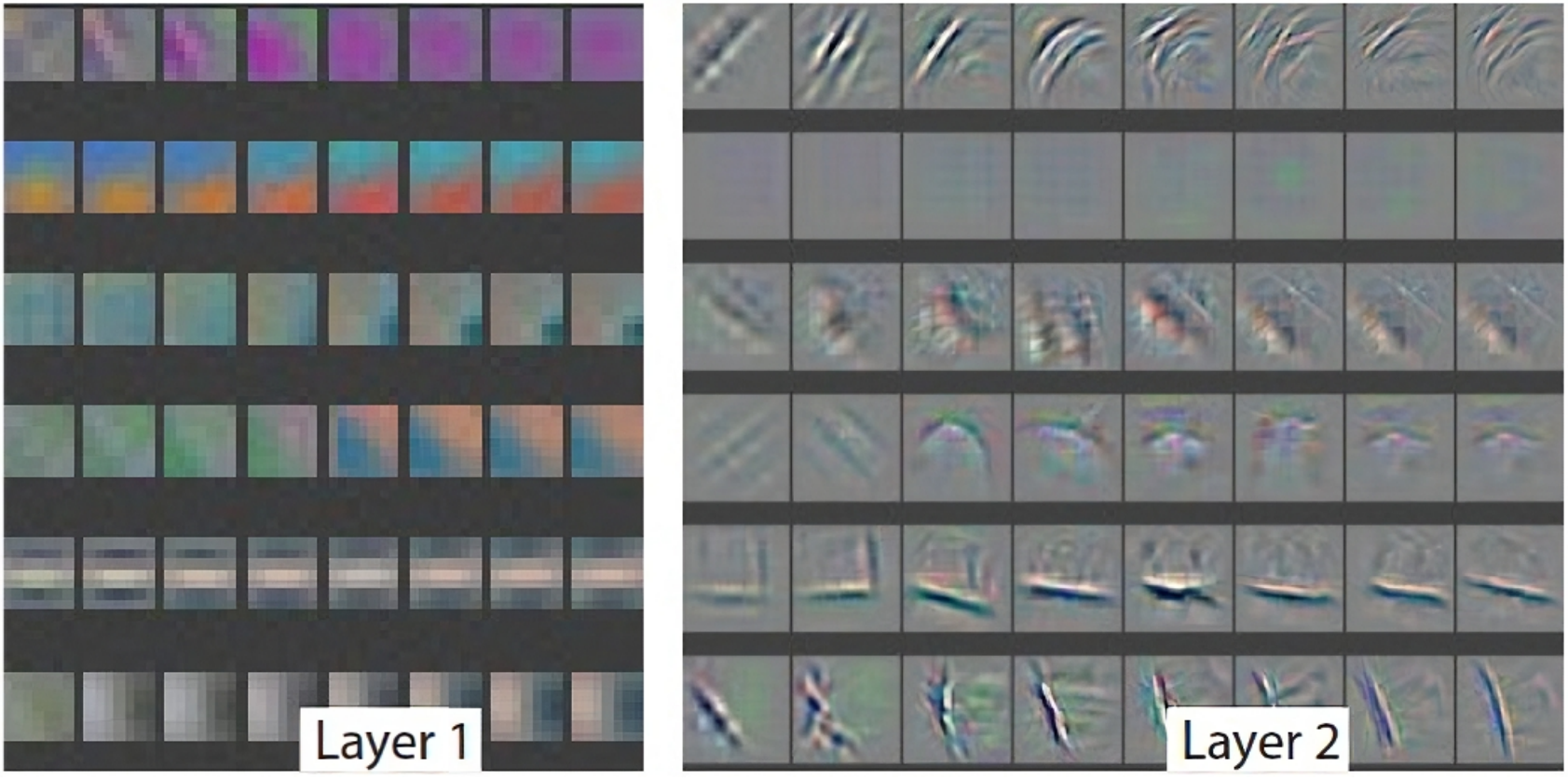


Figure 1: Figure de DeconvNet

Résultats expérimentaux

Exemples de visualisations (2/2)

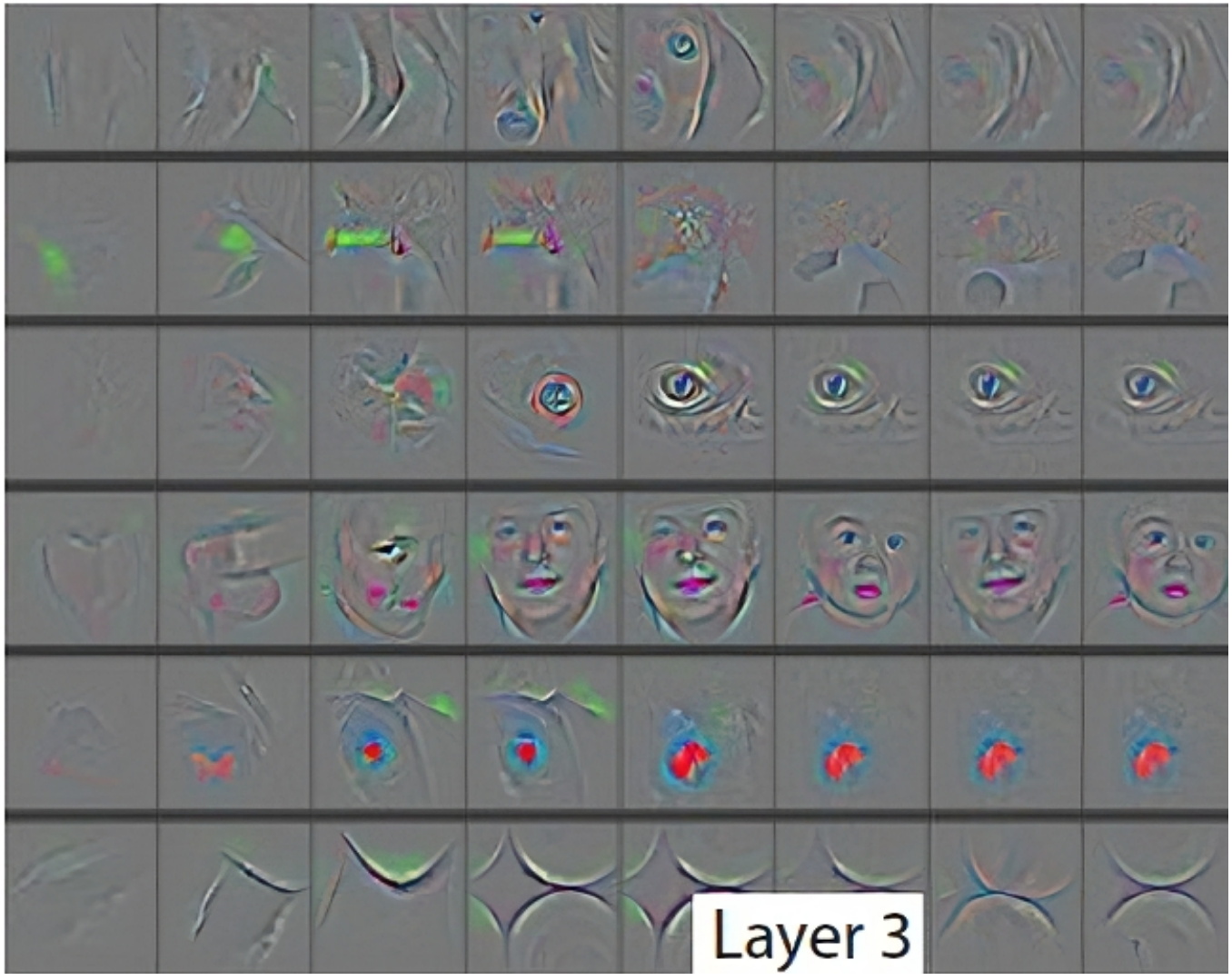
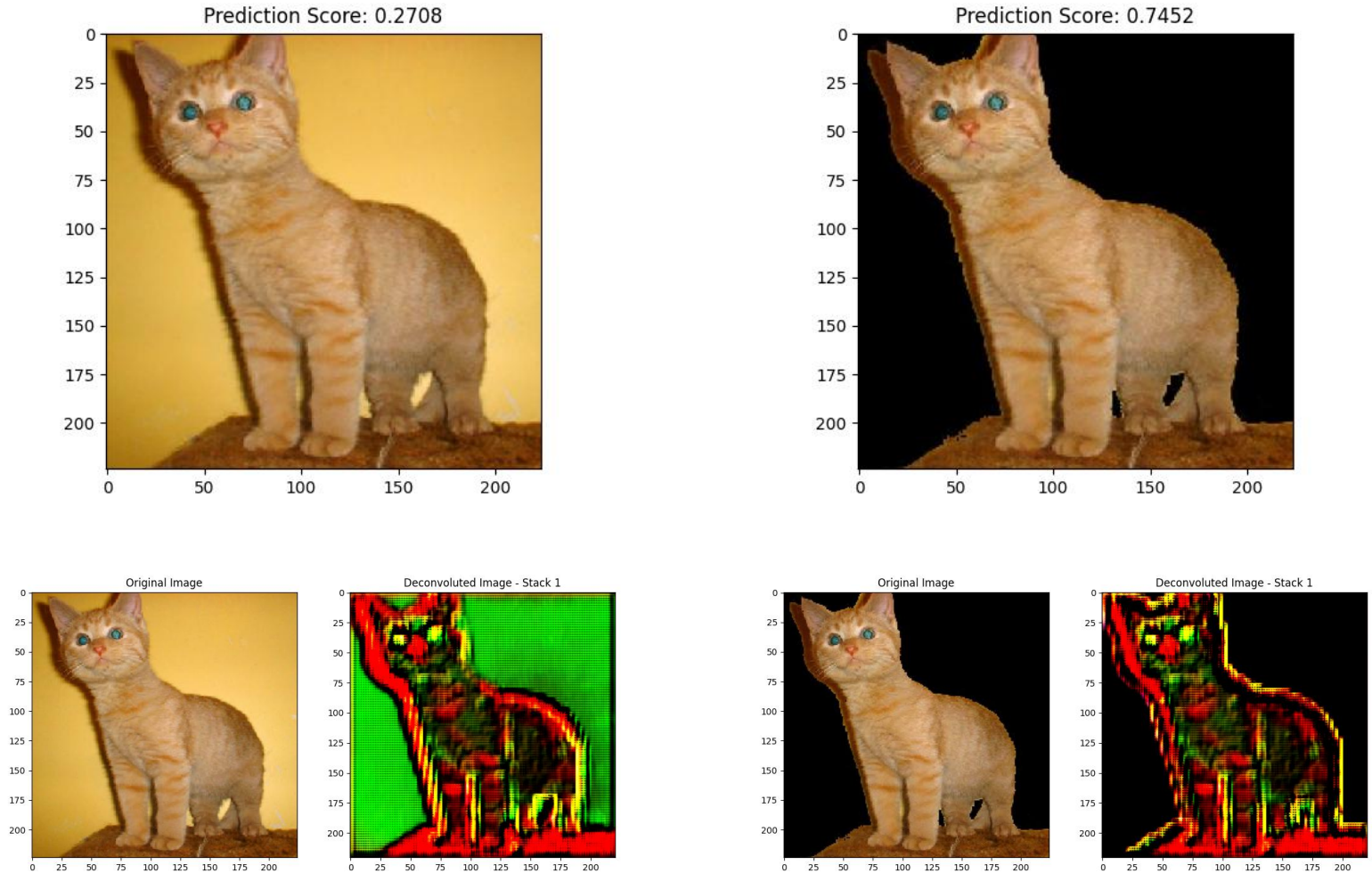


Figure 2: Figure de DeconvNet

Résultats expérimentaux
Simulation deconvolution (1/2)



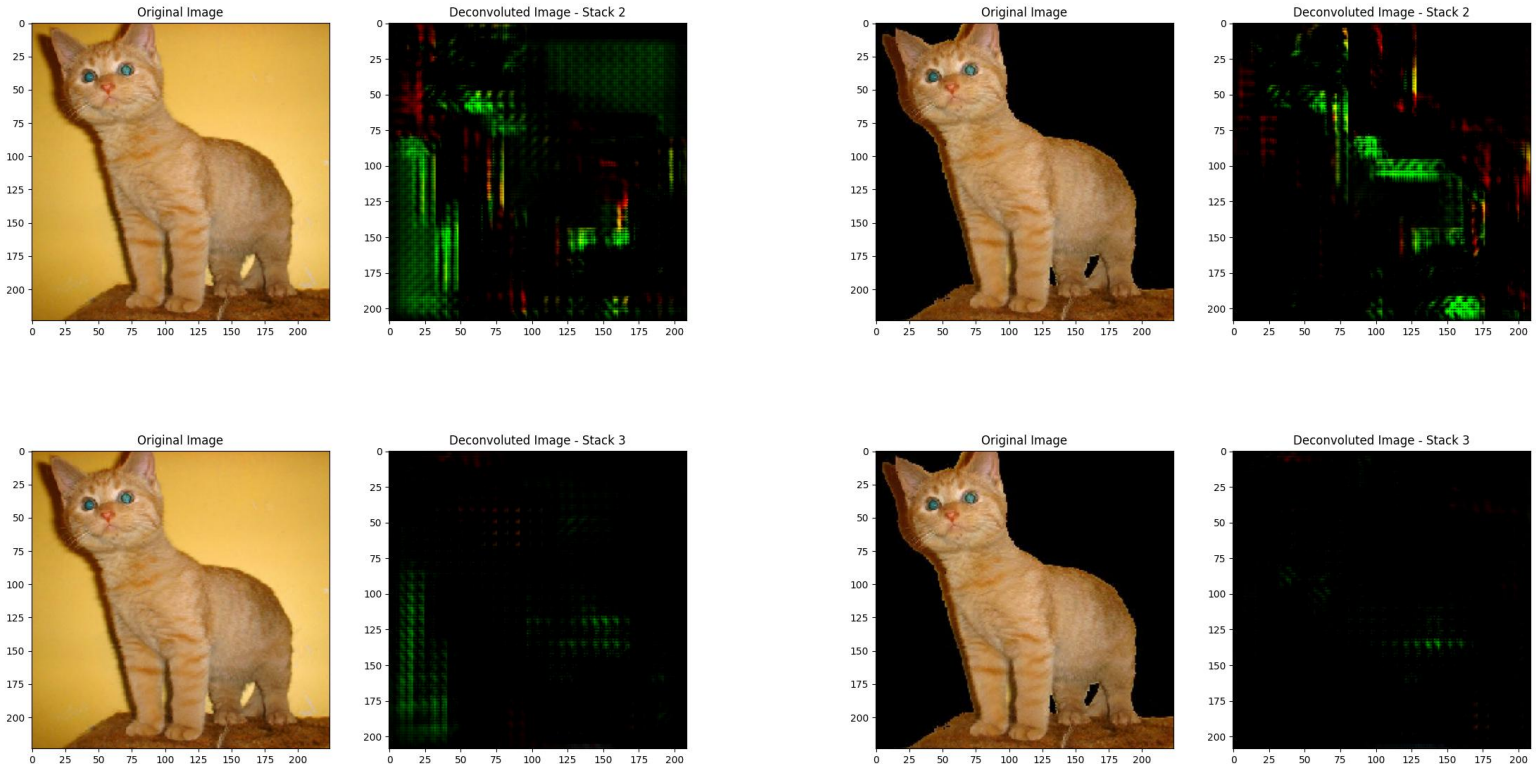
(a) Avec fond de couleur

(b) Sans fond de couleur

Figure 3: Predictions initials et couche 1

Résultats expérimentaux

Simulation deconvolution (2/2)



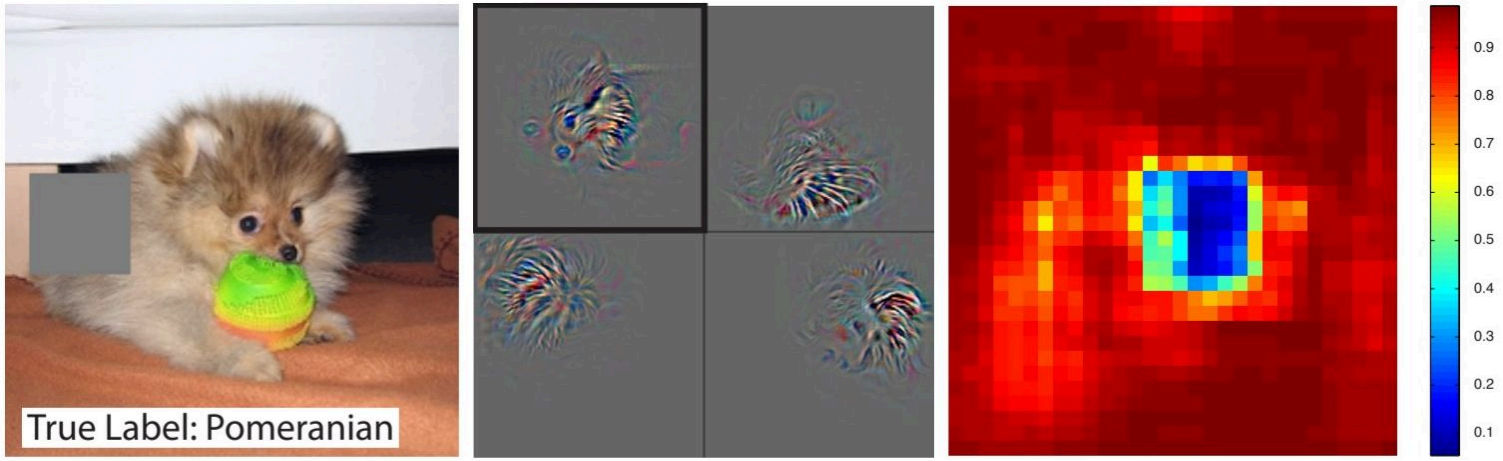
(a) Avec fond de couleur

(b) Sans fond de couleur

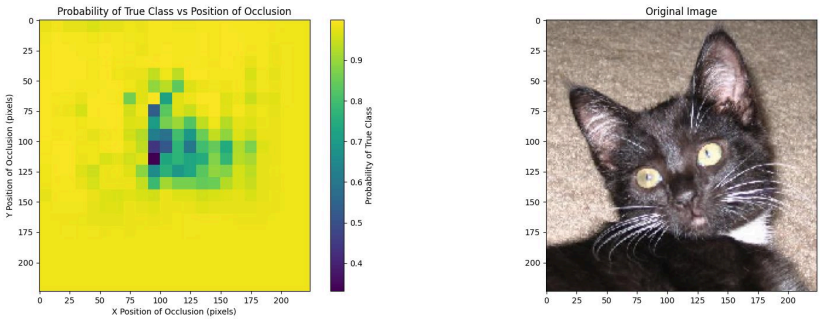
Figure 4: Couches 2 et 3

Résultats expérimentaux

Visualisation des activations (Heatmap)



(a) Résultat de l'article original : Image d'entrée, Projection de la carte de caractéristiques la plus forte, Probabilité de la classe correcte.

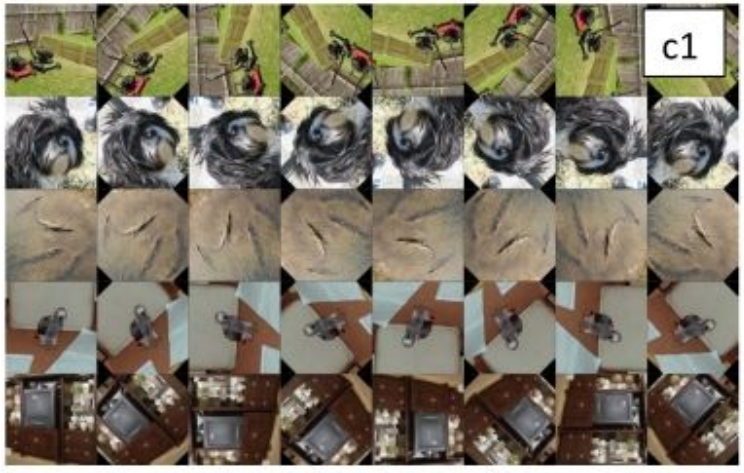


(b) Résultat de la simulation reproduite

Figure 5: Impact de l'occultation sur les activations des caractéristiques et les prédictions de classe

Résultats expérimentaux

Effet de la rotation sur les performances du modèle



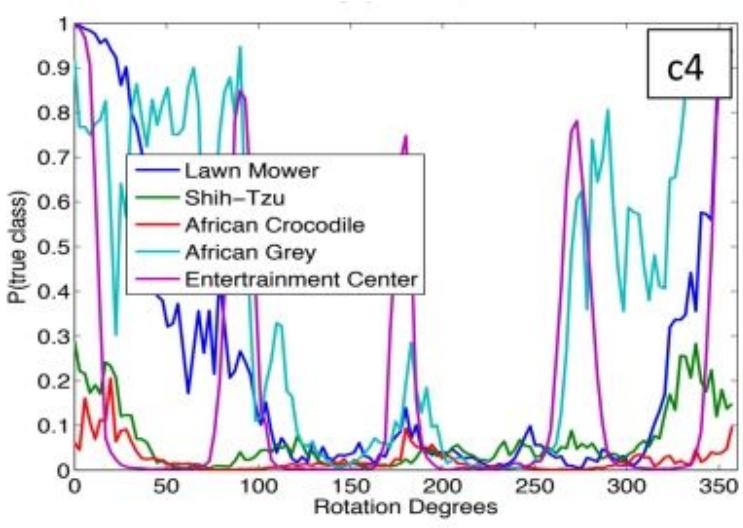


Figure 6: Effet de la rotation sur la classification

Conclusion

Démystification des CNN

Introduction du DeconvNet

Analyse des Performances des Modèles

Avancé de la Recherche sur l'Interprétabilité

Supplément

Le cadre statistique(1/3)

- **Apprentissage supervisé pour la classification d'images :** Le problème est formulé dans un cadre probabiliste, où l'objectif est de modéliser la distribution conditionnelle $P(Y \mid X)$, où X représente les images et Y les étiquettes de classes correspondantes. Le but est de trouver une fonction f_θ (paramétrée par les poids θ du réseau) qui approxime cette distribution pour prédire correctement la classe d'une image donnée.
- **Minimisation de la fonction de perte :** La fonction de perte couramment utilisée est la **cross-entropy**, qui mesure la divergence entre la distribution des probabilités prédites par le modèle et la distribution réelle (étiquettes). La minimisation de cette perte correspond à la maximisation de la **vraisemblance** des données sous le modèle paramétré par θ. En termes probabilistes, il s'agit de maximiser $P(Y \mid X, \theta)$ pour l'ensemble des données d'entraînement.

Supplément

Le cadre statistique(2/3)

- **Optimisation par descente de gradient :** Pour trouver les paramètres θ qui minimisent la perte, l'algorithme de **descente de gradient** est utilisé, avec des variantes comme **SGD** (Stochastic Gradient Descent) qui évalue le gradient sur des mini-lots (batches) de données. Cela permet de mettre à jour les poids du réseau de manière itérative en suivant le gradient de la fonction de perte, pour converger vers un optimum local.
- **Entraînement et validation** : La séparation entre données d'entraînement et de validation peut être vue dans une perspective bayésienne, où les données d'entraînement forment la base de la vraisemblance et les données de validation permettent d'estimer la capacité de généralisation du modèle, c'est-à-dire de prédire $P(Y \mid X)$ pour de nouvelles images.

Supplément

Le cadre statistique(3/3)

- **Prédictions et incertitude :**
  - Le modèle CNN calcule une distribution de probabilité sur les classes pour chaque image, interprétée comme une approximation de $P(Y \mid X, \theta)$. L'étiquette de classe prédite correspond à la **classe ayant la probabilité maximale** (MAP : Maximum A Posteriori).
  - L'analyse des prédictions sur les images de validation permet de mieux comprendre l'incertitude du modèle, notamment lorsque les probabilités prédites sont réparties entre plusieurs classes, ce qui peut indiquer une **incertitude sur les caractéristiques visuelles**.