Alexandre Martin

Etude d'article :

Learning methods for generic object recognition with invariance to pose and lighting

Y. LeCun; Fu Jie Huang; L. Bottou

Sommaire

1. Introduction

L'article de LeCun, Huang et Bottou est une étude des performances réalisées sur différentes méthodes de classification vis-à-vis problème de la reconnaissance d'objets dans des images, invariablement de leur position et de la luminosité, à partir du jeu de données NORB.

Les auteurs décrivent d'abord ce jeu de données et son utilisation, avant d'exhiber les résultats donnés par les différentes méthodes de classification.

Leurs objectifs sont les suivants :

- décrire le jeu de données de reconnaissance d'objet le plus large (de l'époque)
- exhiber la performance des méthodes standards
- comparer les méthodes plus complexes (K-plus proches voisins, machine à vecteur de support, réseau de convolution)
- évaluer la performance des méthodes empiriques lorsque la taille du problème augmente
- mesurer dans quelle mesure les méthodes d'apprentissage peuvent apprendre de la variabilité des images d'objets en 3D
- déterminer si une entrée binoculaire apporte un avantage pour la reconnaissance

2. Données, expériences et résultats

2.1. Description des données

2.1.1. Images originelles : le jeu de données NORB

Le jeu de données NORB est constitué d'images de 50 figurines, divisées en cinq catégories : voitures, avions, camions, humains et animaux. Celles-ci sont photographiées selon 9 élévations, 36 azimuts et 6 conditions d'éclairage différentes par deux caméras faiblement espacées (pour simuler la vision humaine) : on obtient ainsi 194 400 images de taille 640x480, soit un total de 179 Go de données. A noter que les images sont toutes exprimées en nuances de gris et que les figurines ont été peintes pour éviter les effets de texture : le seul indice qui doit être pris en compte est la forme de l'objet.

Dans toute la suite, les images associées aux 5 premières figurines de chaque catégorie forment le jeu d'entraînement, tandis que les autres forment le jeu de test. On séparera d'ailleurs 2 cas :

- cas monoculaire : toutes les images sont utilisées telle quelle
- cas binoculaire : on labellise les images en oeil droit/oeil gauche, l'entraînement et le test seront alors effectués à partir de couples d'images

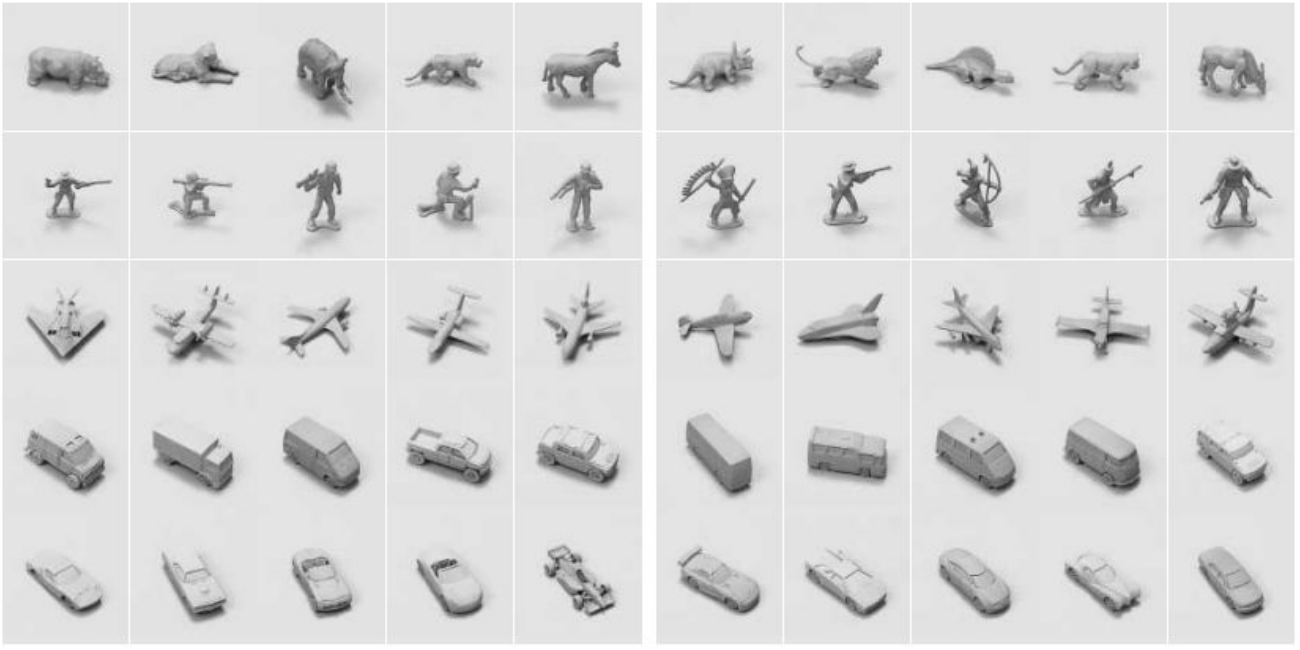

Figure 1 : Ensemble des figurines utilisées pour générer le NORB (entraînement/test)

2.1.2. Images modifiées

Les auteurs utilisent quatre jeu de données dérivés du NORB pour effectuer leurs expériences. Les deux premiers jeux de données seront utilisés pour résoudre des problème de catégorisation pure (peu réaliste en pratique) :

- le jeu de données *normalisé* : c'est une fraction du jeu NORB
- le jeu de données *nerveux* : les images du jeu normalisé sont légèrement modifiées (position de l'objet, échelle, luminosité, …) aléatoirement, produisant ainsi chacune plusieurs nouvelles images

Dans les deux autres jeux, les objets du NORB sont découpés et placés sur des arrières-plans. Il est alors nécessaire d'ajouter une nouvelle classe correspondant aux arrières-plans vides d'objet. Ces jeux seront plutôt utilisés dans des problèmes de reconnaissance :

- le jeu de données *texturé* : les images sont perturbées comme dans le jeu nerveux, de plus l'arrière-plan présente des disparités régulières
- le jeu de données *encombré* : les images sont perturbées comme dans le jeu nerveux, de plus l'arrière-plan présente des disparités irrégulières

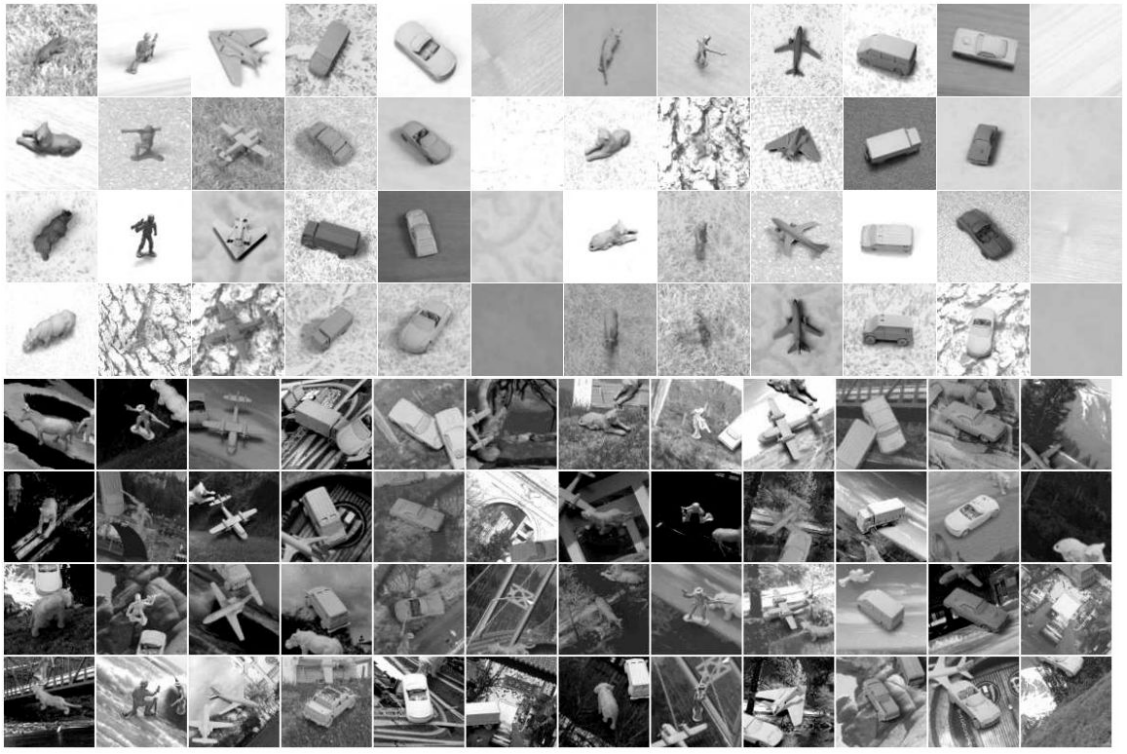

Figure 2 : Images des jeux *texturé* (4 premières rangées) et *encombré* (4 dernières)

2.2. Méthodes de classification comparées

2.2.1. K-plus proches voisins

Il s'agit d'une méthode non-paramétrique qui mémorise l'ensemble des observations durant l'apprentissage pour la classification. Pour prédire la classe d'une nouvelle entrée, l'algorithme recherche ses K plus proches voisins, et lui associe la classe qui en sort majoritaire.

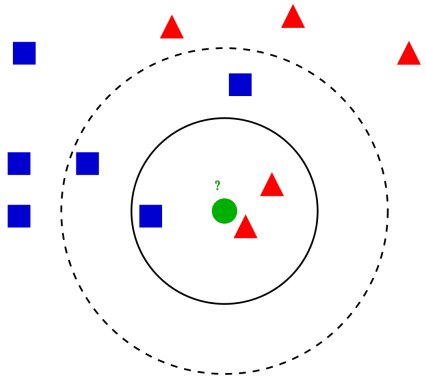

Figure 3 : Pour K = 3, on obtient un triangle rouge ; pour K = 5, un carré bleu

Les meilleurs résultats sont ici obtenus pour K = 1.

2.2.2. Machine à vecteurs supports

Il s'agit d'une classe d'algorithmes d'apprentissage reposant sur la recherche de l'hyperplan de marge optimale, c'est-à-dire qui classe ou sépare correctement les données tout en étant le plus éloigné possible de toutes les observations. Une nouvelle entrée est alors associée à une classe selon sa position vis-à-vis de cet hyperplan.

Cette méthode sera ici appliquée avec des noyaux gaussiens.

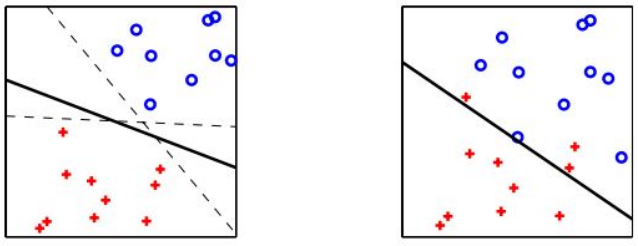

Figure 4 : Deux exemples pour un hyperplan de dimension 1

2.2.3. Réseaux de convolution

Un réseau de convolution est un type de réseau de neurones ayant tendance à repérer facilement des motifs récurrents. Quand on lui présente une nouvelle image, le réseau de convolution ne sait a priori si les caractéristiques seront présentes dans l'image ou où elles pourraient être, il cherche donc à les trouver dans toute l'image et dans n'importe quelle position : cela augmente sa robustesse.

On utilisera ici un réseau à 6 couches.

2.3. Résultats

K-NN = K-plus proches voisins
SVM Gauss = Machine à vecteurs supports, noyau gaussien
Conv Net = réseau de convolution
norm-unif = jeu normalisé
jitt-unif = jeu nerveux
jitt-text = jeu texturé
jitt-clutt = jeu encombré
raw 1 = cas monoculaire
raw 2 = cas binoculaire
PCA-95 = composantes de l'analyse en composantes principales des images

2.3.1. Sur les images sans arrière-plan

Classification				
exp#	Classifier	Input	Dataset	Test Error
1.0	Linear	raw 2x96x96	norm-unif	30.2%
1.1	K-NN (K=1)	raw 2x96x96	norm-unif	18.4 %
1.2	K-NN (K=1)	PCA 95	norm-unif	16.6%
1.3	SVM Gauss	raw 2x96x96	norm-unif	N.C.
1.4	SVM Gauss	raw 1x48x48	norm-unif	13.9%
1.5	SVM Gauss	raw 1x32x32	norm-unif	12.6%
1.6	SVM Gauss	PCA 95	norm-unif	13.3%
1.7	Conv Net 80	raw 2x96x96	norm-unif	6.6%
1.8	Conv Net 100	raw 2x96x96	norm-unif	6.8%
2.0	Linear	raw 2x96x96	jitt-unif	30.6%
2.1	Conv Net 100	raw 2x96x96	jitt-unif	7.1%

- La méthode des plus proches voisins (force brute) fonctionne, cependant elle nécessite des ressources beaucoup trop élevées en mémoire et en calcul
- La méthode à vecteurs supports n'est plus applicable lorsque la taille et la complexité du problème deviennent trop importantes : plusieurs jours de calcul ne sont pas suffisants
- Les réseaux de convolution présentent les meilleurs résultats (moins de 7% d'erreur pour une vue binoculaire). De plus, ceux-ci sont robustes à de faibles perturbations.

2.3.2. Sur les images avec arrière-plan

Detection/Segmentation/Recognition				
exp#	Classifier	Input	Dataset	Test Error
5.1	Conv Net 100	raw 2x96x96	jitt-text	10.6%
6.0	Conv Net 100	raw 2x96x96	jitt-clutt	16.7%
6.2	Conv Net 100	raw 1x96x96	jitt-clutt	39.9%

- La plupart des erreurs sont des objets classifiés comme arrières-plans et des voitures classifiées comme des camions
- La différence est importante entre le cas monoculaire et le cas binoculaire : ce dernier va ici apporter de la robustesse au modèle en permettant de localiser la frontière de l'objet d'une manière plus nette.

2.3.3. Sur des images réelles

Le réseau de convolution entraîné sur le jeu de données encombré est suffisamment robuste pour reconnaître des objets réels de façon satisfaisante.

Figure 5 : Exemples de classification

3. Lien avec DeepSpeech

Il est possible de faire une analogie avec le problème étudié dans DeepSpeech en faisant les assimilations suivantes :

image	extrait audio
objet	mot
pixel	son
arrière-plan	bruit
réseau de convolution	réseau récurrent bidirectionnel

En effet, d'un point de vue informationnel, un réseau de convolution essaye de reconnaître des motifs entre les pixels en utilisant l'information disponible dans un proche rayon spatial. De manière similaire, un réseau de neurones récurrent bidirectionnel, comme utilisé dans DeepSpeech, utilise l'information disponible dans un proche rayon temporel pour reconnaître des motifs liant les différents sons. Un tel réseau de neurones reconnaissant des motifs sera plus robuste dans un contexte de bruit important.

De plus, comme indiqué dans l'article DeepSpeech, celui-ci reprend le processus de création de nouvelle données d'entraînement à partir d'une transformation des données de base de manière aléatoire, ce qui permet de dupliquer le nombre de données disponibles pour l'entraînement (cf 2.1.2.) dans un contexte bruyant.

4. Deux articles d'ouverture

4.1. Learning a Sparse Representation for Object Detection (Agarwal and Roth, 2002)

L'article construit une méthode de représentation capturant un maximum d'information sur les parties d'une image et les relations spatiales entre elles. La notion de fonction d'activation de classification est introduite pour former un détecteur efficace des objets d'une classe.

Cette méthode fonctionne avec succès sur un ensemble d'images correspondant à des vues latérales de voitures, et atteint un taux de détection élevé sur des images du monde réel. Cette approche est extensible à d'autres objets ayant des parties suffisamment distinctes dans une configuration spatiale stable.

L'article décrit la différence entre classification et détection, et délivre une méthode générale pour produire un bon détecteur à partir d'un classificateur. Il met également en exergue l'importance de la standardisation des schémas d'évaluation des objets.

4.2. Appearance-Based Object Recognition Using Multiple Views (Selinger and Nelson, 2001)

L'article présente une méthode de reconnaissance d'objets à partir de vues multiples prises séparément. Cette méthode présente pour avantage de pouvoir être configurée facilement, et d'utiliser également des systèmes de reconnaissance visuelle avec entrée unique.

Une telle méthode permet d'améliorer le niveau de reconnaissance, avec toutefois deux limites importantes : tout d'abord, si la performance d'une seule vue est faible, celle en combinant deux vues le sera également. De plus, imposer des contraintes entre les différentes vues ne produit pas forcément une augmentation sensible de la performance.

5. Conclusion

L'article de LeCun, Huang et Bottou met en lumière les limites des méthodes classiques de reconnaissance d'images lorsque le nombre de données devient très élevées : celles-ci deviennent en effet trop gourmandes en puissance de calcul pour un résultat trop peu précis.

L'entraînement d'un réseau de convolution permet d'obtenir une structure de classification qui soit à la fois robuste à la quantité et à la qualité des données d'entrée. C'est de plus la structure la plus précise ici : elle peut également être utilisée pour la reconnaissance d'images réelles.