Apprentissage et exploitation de représentations sémantiques pour la classification et la recherche d'images

Maxime Bucher

Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité INFORMATIQUE

Préparée au sein de l'Université de Caen Normandie

Apprentissage et exploitation de représentations sémantiques pour la classification et la recherche d'images

Présentée et soutenue par
Maxime BUCHER

Thèse soutenue publiquement le 27/11/2018 devant le jury composé de		
Mme CÉLINE HUDELOT	Professeur des universités, 35 SUPELEC	Rapporteur du jury
M. NICOLAS THOME	Professeur des universités, Conservatoire Nal des Arts et Métiers	Rapporteur du jury
M. STÉPHANE HERBIN	Maître de conférences, ONERA	Membre du jury
M. FREDERIC PRECIOSO	Professeur des universités, UNIVERSITE NICE SOPHIA ANTIPOLIS	Président du jury
M. FREDERIC JURIE	Professeur des universités, UNIVERSITE CAEN NORMANDIE	Directeur de thèse

Thèse dirigée par FREDERIC JURIE, Groupe de recherche en informatique, image, automatique et instrumentation

UNIVERSITÉ CAEN NORMANDIE

MIIS

GREYC

Abstract

In this thesis, we examine some practical difficulties of deep learning models. Indeed, despite the promising results in computer vision, implementing them in some situations raises some questions.

For example, in classification tasks where thousands of categories have to be recognised, it is sometimes difficult to gather enough training data for each category. We propose two new approaches for this learning scenario, called «zero-shot learning». We use semantic information to model classes which allows us to define models by description, as opposed to modelling from a set of examples. In the first chapter we propose to optimize a metric in order to transform the distribution of the original data and to obtain an optimal attribute distribution. In the following chapter, unlike the standard approaches of the literature that rely on the learning of a common integration space, we propose to generate visual features from a conditional generator. The artificial examples can be used in addition to real data for learning a discriminant classifier.

In the second part of this thesis, we address the question of computational intelligibility for computer vision tasks. Due to the many and complex transformations of deep learning algorithms, it is difficult for a user to interpret the returned prediction. Our proposition is to introduce what we call a «semantic bottleneck» in the processing pipeline, which is a crossing point in which the representation of the image is entirely expressed with natural language, while retaining the efficiency of numerical representations. This semantic bottleneck allows to detect failure cases in the prediction process so as to accept or reject the decision.

Keywords

zero-shot learning, attributes, embedding, metric learning, semantic bottlenecks, retrieval

Résumé

Dans cette thèse nous étudions différentes questions relatives à la mise en pratique de modèles d'apprentissage profond. En effet malgré les avancées prometteuses de ces algorithmes en vision par ordinateur, leur emploi dans certains cas d'usage réels reste difficile.

Une première difficulté est, pour des tâches de classification d'images, de rassembler pour des milliers de catégories suffisamment de données d'entraînement pour chacune des classes. C'est pourquoi nous proposons deux nouvelles approches adaptées à ce scénario d'apprentissage, appelé «classification zero-shot». L'utilisation d'information sémantique pour modéliser les classes permet de définir les modèles par description, par opposition à une modélisation à partir d'un ensemble d'exemples, et rend possible la modélisation sans donnée de référence. L'idée fondamentale du premier chapitre est d'obtenir une distribution d'attributs optimale grâce à l'apprentissage d'une métrique, capable à la fois de sélectionner et de transformer la distribution des données originales. Dans le chapitre suivant, contrairement aux approches standards de la littérature qui reposent sur l'apprentissage d'un espace d'intégration commun, nous proposons de générer des caractéristiques visuelles à partir d'un générateur conditionnel. Une fois générés ces exemples artificiels peuvent être utilisés conjointement avec des données réelles pour l'apprentissage d'un classifieur discriminant.

Dans une seconde partie de ce manuscrit, nous abordons la question de l'intelligibilité des calculs pour les tâches de vision par ordinateur. En raison des nombreuses et complexes transformations des algorithmes pro-

fonds, il est difficile pour un utilisateur d'interpréter le résultat retourné. Notre proposition est d'introduire un «goulot d'étranglement sémantique» dans le processus de traitement. La représentation de l'image est exprimée entièrement en langage naturel, tout en conservant l'efficacité des représentations numériques. L'intelligibilité de la représentation permet à un utilisateur d'examiner sur quelle base l'inférence a été réalisée et ainsi d'accepter ou de rejeter la décision suivant sa connaissance et son expérience humaine.

Mots-clés

classification zero-shot, attributs, plongement sémantique, apprentissage de métrique, goulot d'étranglement sémantique, recherche d'images

Table des matières

Page

Liste des tableaux **ix**

Table des figures **xi**

Introduction **1**

1 Apprentissage d'un espace sémantique optimal pour la reconnaissance visuelle sans exemple d'apprentissage **13**

1.1 Introduction 14

1.2 État de l'art 20

1.2.1 Création d'un ensemble d'attributs 20

1.2.2 Utilisation d'attributs comme représentation intermédiaire 21

1.2.3 Plongement de l'image et de son l'étiquette dans un espace sémantique commun 23

1.2.4 Reconnaissance visuelle avec un nombre réduit d'exemples 25

1.2.5 Apprentissage de métrique et sélection de paires optimales 26

1.3 Apprentissage de métrique pour un espace d'attribut optimal 28

1.3.1 Classification zero-shot 28

1.3.2 Score de compatibilité 29

1.3.3 Plongement dans un espace d'attributs 31

1.3.4 Apprentissage de métrique 32

1.3.5 Méthodes intelligentes de sélection des paires négatives . . . 34
1.3.6 Prise en compte du déséquilibre entre paires positives et négatives . . . 37
1.3.7 Reconnaissance et recherche d'image . . . 38
1.4 Expériences . . . 40
1.4.1 Jeux de données . . . 40
1.4.2 Implémentation . . . 42
1.4.3 Classification zero-shot . . . 43
1.4.4 Classification avec un faible nombre d'exemples de référence . . . 48
1.4.5 Recherche d'images zero-shot . . . 50
1.5 Conclusion . . . 53

2 Génération de caractéristiques visuelles conditionnée par représentation sémantique 55
2.1 Introduction . . . 56
2.2 État de l'art . . . 59
2.2.1 Génération d'exemples de référence . . . 59
2.2.2 De l'approche transductive à généralisée . . . 61
2.2.3 Informations sémantiques hétérogènes . . . 63
2.3 Génération de caractéristiques visuelles pour l'ensemble des classes non vues . . . 66
2.3.1 Approche discriminante . . . 66
2.3.2 Génération de caractéristiques visuelles . . . 67
2.3.3 Generative Moment Matching Network . . . 68
2.3.4 Wasserstein Generative Adversarial Network . . . 71
2.3.5 Denoising Auto-Encoder . . . 72
2.3.6 Adversarial Auto-Encoder . . . 74
2.3.7 Classification de caractéristiques générées . . . 75
2.4 Expériences . . . 77

2.4.1 Jeux de données 77
2.4.2 Implémentation 78
2.4.3 Classification zero-shot 79
2.4.4 Classification zero-shot généralisée 84
2.4.5 Classification zero-shot à grande échelle 88
2.5 Conclusion 91

3 Goulot d'étranglement sémantique pour la détection de défaillance 93
3.1 Introduction 94
3.2 État de l'art 97
3.2.1 Extraction d'informations sémantiques 97
3.2.2 Transfert d'informations entre domaines 99
3.2.3 Produire des représentations intelligibles 99
3.2.4 Évaluer une explication 101
3.2.5 Générer des questions discriminantes 102
3.3 Génération de représentations sémantiques par un système de VQA 104
3.4 Goulot d'étranglement sémantique 108
3.4.1 Représentations vectorielles, encodeurs, décodeurs . 109
3.4.2 Générateur de description 113
3.4.3 Générateur de réponse 114
3.4.4 Génération de questions discriminantes 116
3.4.5 Encodeur de représentation sémantique 117
3.4.6 Apprentissage du modèle 118
3.5 Expériences 120
3.5.1 Recherche d'images 122
3.5.2 Classification multi-étiquettes 123
3.5.3 Analyse du goulot d'étranglement sémantique . . . 125
3.5.4 Modification de la représentation sémantique 130
3.5.5 Détection de défaillance 130

3.6 Conclusion 135

Conclusion et Perspectives **137**

Publications **143**

Bibliographie **145**

Liste des tableaux

Tableau **Page**

1.1 Statistiques sur les jeux de données. 40

1.2 Précision multi-classe en classification zero-shot pour notre modèle MLZSL. Cont. : plongement de l'image dans l'espace initial d'attribut, Métr. : transformation appliquée à l'espace initial d'attribut, Sélect. : stratégie de sélection de paires négatives suivant l'incertitude et la corrélation. 43

1.3 Précision zero-shot sur le jeu de données CUB en fonction du ratio paire négative/positive et la méthode de sélection de paires utilisée. 45

1.4 Précision multi-classes en classification zero-shot. Les performances de la première partie de tableau proviennent de l'article [163]. 47

1.5 Recherche d'image zero-shot : précision moyenne (%) sur 4 jeux de données. Les caractéristiques visuelles sont extraites d'un réseau VGG19 [143] à la couche fc7. Pour une comparaison objective, nous suivons les mêmes sous-ensembles que Zhang [173]. 50

2.1 L'apprentissage zero-shot se présente suivant différentes configurations. X : image Y_s : ensemble des classes vues, Y_u : ensemble des classes non vues, $\{X_{i:n}\}_u$: ensemble d'images non étiquetées. 61

2.2 Statistiques sur le jeu de données ImageNet [39]. Y : nombre de classe total, Y^s : nombre de classe vue, Y^s : nombre de classe non vue. *2-hop*, *3-hop* et *All* correspondent aux différents sous-ensembles de test. 77
2.3 Précision en classification zero-shot. Les performances de la première partie de tableau proviennent de l'article [163]. . . . 80
2.4 Classification zero-shot généralisée. u est la précision de classification des images de test de classes non vues, s est la précision de classification des images de test des classes vues et H indique la moyenne harmonique entre u et s. 85
2.5 Classification zero-shot à grande échelle sur l'ensemble de données ImageNet. Nous reportons la précision moyenne pour différents scénarios (sous ensembles et généralisé). Les caractéristiques visuelles sont extraites d'un réseau GoogLeNet [148]. 89

3.1 Score NDCG pour la tâche de recherche d'images. Performance / Air sous la courbe (AUC) pour différentes valeurs de R. 121
3.2 Recherche d'images. Scores NDCG / AUC après avoir supprimé certains composants du modèle. 121
3.3 Précision moyennée par classe pour la baseline et les différents composants de notre modèle. 123
3.4 Pourcentage d'apparition pour les deux tâches. La colonne «classes» correspond au pourcentage d'occurrence de l'ensemble des étiquettes du problème de classification multi-labels. . . . 128
3.5 Statistiques de prédiction de défaillance. 132
3.6 Classification multi-labels. 133

Table des figures

Figure **Page**

0.1 *Image extraite de [4]* : les différentes tâches de vision par ordinateur. 1

0.2 *Image extraite de [9]* : l'objectif de la tâche consiste à répondre à une question posée sur une image, en langage naturel. . . . 2

0.3 *Image extraite de [1]* : les espèces d'oiseaux communes sont représentées par des images facilement disponibles, contrairement aux classes des espèces rares qui sont constituées de seulement quelques exemples. 7

0.4 *Image extraite de [5] :* visualisation des couches intermédiaires d'un réseau convolutif. 8

0.5 *Image extraite de [136] :* les cartes de saillance générées permettent de visualiser les zones de l'image utilisées pour la prise de décision. 9

1.1 *Image extraite de [81] :* données médicales provenant d'un système de mammographie. Détection automatique de la nature cancérigène du kyste. Ce type d'approche à un coût très élevé notamment dans l'acquisition de l'appareil ainsi qu'en temps et effort humain. 15

1.2 *Image extraite de [3]* : l'ensemble des catégories possibles est potentiellement infini, ce qui rend difficile la collecte et l'annotation de données pour toutes les classes (de plus de nouvelles catégories émergent de manière ininterrompue). 16

1.3 *Image extraite de [2] Segway : véhicule électrique monoplace, constitué d'une plateforme, muni de deux roues parallèles et d'un manche de maintien et de conduite sur laquelle l'utilisateur se tient debout.* Inspirée par les capacités humaines, notre approche permet de reconnaître un objet à partir d'une représentation sémantique. 17

1.4 L'utilisation d'attributs permet de construire un nouvel espace de représentation pour l'apprentissage. Chaque classe est représentée par un vecteur d'attributs correspondant à la présence ou l'absence de ceux-ci. Elle permet de définir les modèles par description, par opposition à une modélisation à partir d'un ensemble d'exemples, et rend possible la modélisation sans donnée de référence («zero-shot learning»). 18

1.5 *Figure extraite de [121] :* introduit une approche pour définir un vocabulaire d'attributs à la fois humainement compréhensible et discriminant. Le système prend en entrée des images étiquetées par objet/scène et renvoie en sortie un ensemble d'attributs obtenus par interactions avec un annotateur humain, chargé de distinguer les bons aux mauvais attributs. 20

1.6 *Figure extraite de [84]* : (a) par le biais de détecteurs d'attributs, le modèle extrait la représentation de l'image $\mathbf{x}$. Chaque label d'entraînement y_i est représenté par la présence ou l'absence d'attributs a_i. Les relations classe-attribut sont fixes (lignes épaisses). Au moment du test, les attributs prédits à partir de l'image permettent d'inférer la classe, même pour les classes non vues. (b) Les attributs $\mathbf{a}$ forment une couche de connexion intermédiaire entre les classes vues y et non vues z. Les attributs ne sont plus prédits à partir de l'image, mais des vraisemblances de celle-ci d'appartenir aux classes vues. . . . 22

1.7 *Image extraite de [6]* : $\theta(\mathbf{x})$ correspond aux caractéristiques de l'image $\mathbf{x}$. Les étiquettes de classes y sont représentées par un ensemble d'attributs $\phi(y)$. La fonction de score est $F(\mathbf{x}, y; \mathbf{W})$. . 24

1.8 *Image extraite de [86] :* localise un lieu à partir d'une image, ainsi qu'à partir de ses transformations géométriques et photométriques. 26

1.9 À l'aide d'un réseau convolutif, l'image est plongée dans l'espace sémantique (attributs verts). Certains attributs sont difficiles à détecter, redondants ou encore inutiles. La métrique appliquée à l'image et à la représentation de la classe, est capable à la fois de sélectionner et de transformer la distribution de données originales (attributs bleus). 28

1.10 Effet de la métrique appliquée à un espace de représentation non optimal. La métrique diminue la distance entre les paires positives (ronds bleus) et maximise la distance entre les paires négatives (carré rouge). 32

1.11 Différentes stratégies de sélection des paires négatives. Les symboles orange représentent celles potentiellement sélectionnées par la stratégie. *Incertitude* : sélectionne les exemples proches ou du mauvais côté de la frontière de décision. *Corrélation* : les exemples représentant au mieux une classe sont sélectionnés. *Incertitude*/*Corrélation* : combine les deux approches, les exemples proches de la frontière de décision ont plus de chance d'être sélectionnés. 35

1.12 Exemples d'images et d'attributs utilisés pour l'évaluation. Chaque jeu de données possède son propre ensemble d'attributs. 41

1.13 Précision zero-shot en fonction de la dimension de la métrique. Les meilleurs résultats sont obtenus quand la dimension est réduite de 40% par rapport à l'espace d'attributs initial. Les scores avec et sans contrainte de plongement sont reportés. . . 45

1.14 Évolution de la performance sur les 3 jeux de données CUB, P&Y et SUN en fonction du nombre d'itérations d'entraînement pour 1 et 100 de ratio. La sélection intelligente de paires négatives permet d'accélérer la convergence du modèle. 46

1.15 Performance de classification avec un faible nombre d'exemples de référence. Précision moyenne (%) en fonction du nombre d'exemples d'apprentissage provenant des classes non vues. . . 49

1.16 Résultats qualitatifs sur l'ensemble de données CUB. Les classes les plus incertaines y sont reportées. 50

1.17 Courbes de précision rappel. 52

2.1 *Figure extraite de [163] :* génération de caractéristiques visuelles pour les classes non vues grâce à un «Generative Adversarial Network», conditionné par une représentation d'attributs. Un critère de classification sur les caractéristiques générées permet d'affiner le résultat. 59

2.2 *Figure extraite de [7] :* exemples de descriptions textuelles de l'ensemble de données CUB [154] et FLOWER [116]. 63

2.3 Gauche (*figure extraite de [110]*) : exemple de résultat obtenu avec une approche «word2vec», la sémantique y est respectée. Droite : article Wikipédia utilisé pour représenter sémantiquement la classe zèbre. 64

2.4 Projection t-SNE 2D [99] de la représentation d'attributs (gauche) et des caractéristiques visuelles extraites d'un réseau convolutif ResNet [65] (droite). Cette figure confirme notre intuition, l'espace des caractéristiques visuelles est mieux organisé, plus structuré. 66

2.5 La génération de caractéristique image $\widehat{\mathbf{x}}$ est conditionnée par : la représentation sémantique $\mathbf{a}$ et un échantillon aléatoire d'une distribution multivariée $\mathbf{z}$. Après génération, un modèle de classification $\hat{y} = f_D(\mathbf{x}; \widehat{\mathscr{D}}_u, \mathscr{D}_s)$ peut être optimisé avec supervision pour les classes vues et non vues. 68

2.6 Architecture des modèles génératifs proposés. $\mathbf{z}$: bruit gaussien, $\mathbf{a}$: description sémantique, FC + lrelu : couche entièrement connectée suivie d'une fonction non linéaire type «Leaky ReLUs» [98], $\widehat{\mathbf{x}}$: caractéristiques visuelles générées. 69

2.7 GMMN - classifieur discriminant (softmax) vs fonction de similarité (MLZSL) sur la tâche de classification zero-shot. 81

2.8 Visualisation T-SNE des caractéristiques visuelles générées par le GMMN, C-WGAN, AE et ADV-AE. Nous les comparons aux «vraies» caractéristiques visuelles. Les couleurs représentent les différentes classes de l'ensemble de données AWA2. *(à visualiser en version numérique)* 82

2.9 GMMN - Évolution de la performance en fonction du nombre de caractéristiques visuelles générées par classe, sur la tâche de classification zero-shot. 83

2.10 GMMN - classifieur discriminant vs fonction de similarité sur la tâche de classification zero-shot généralisée 86

2.11 Résultats qualitatifs pour le modèle GMMN. La proximité entre les représentations d'attributs influe sur la capacité de discrimination du modèle. 87

3.1 Goulot d'étranglement sémantique : les images sont remplacées par une représentation textuelle riche (description et quiz visuel), pour des tâches telles que la classification multi-étiquettes ou la recherche d'images. 95

3.2 *Image extraite de [37] :* Visual Dialog, un agent conversationnel dialogue avec un humain à propos du contenu visuel de l'image. 98

3.3 *Image extraite de [172] :* étant donné une image d'entrée, l'arbre de décision analyse quelles parties d'objet sont utilisées dans la prise de décision. 100
3.4 *Image extraite de [90] :* exemple de paire d'images ambiguës, avec à la fois de bonnes et de mauvaises questions discriminantes. 102
3.5 Nous proposons d'utiliser un modèle VQA comme base de connaissances pour représenter des images. Chaque dimension de cette représentation correspond à la probabilité qu'une paire question/réponse soit compatible avec l'image. 105
3.6 «deeper LSTM Q + norm I» : (orange) les images sont représentées par la dernière couche cachée d'un réseau VGG-VeryDeep-19 [143]. (vert) Un LSTM à deux couches code les questions. (bleu) Les représentations de la question et de l'image sont fusionnées via une multiplication élément par élément et utilisées pour la prédiction de la réponse. 106
3.7 Schéma fonctionnel de notre approche. 109
3.8 Générateur de description : composé de deux parties (1) une extraction des caractéristiques visuelles de l'image ; (2) la génération mot à mot d'une description textuelle. 113
3.9 Générateur de réponse : à partir de l'image, de la question et de l'historique du quiz visuel, le modèle répond en langage naturel à la question posée. 114
3.10 Génération de questions discriminantes : un ensemble de questions est produit de manière itérative, dans le but d'affiner la représentation sémantique initiale (description). 116
3.11 Module de fusion : fusionne les représentations sémantiques entre elles (description + quiz visuel). Cette représentation est utilisée pour différentes tâches de vision par ordinateur. 117
3.12 Précision en classification multi-labels en fonction du nombre de questions et réponses posées. 124

3.13 Combinaison de descriptions et de dialogues : requête (images en haut à gauche), descriptions générées, dialogues spécifiques à la tâche, images retournées en utilisant la description (premières lignes) et celles données par notre modèle (deuxièmes rangées). Le dialogue permet de détecter des concepts complémentaires importants manqués par la description. 125
3.14 Adapter le dialogue à la tâche : requête (images en haut à gauche), légendes générées, dialogue générique et spécifique à la tâche, images récupérées en utilisant la description et le dialogue générique (premières lignes) et celles données par notre modèle (deuxièmes rangées). 127
3.15 Partie gauche : description incorrecte corrigée par le dialogue. Droite : objets manquants dans la description, mais découverts en posant des questions pertinentes. 127
3.16 Descriptions et dialogues générés pour les tâches de classification et de recherche. 128
3.17 Représentations sémantiques défaillantes. 129
3.18 Modifications de la représentation sémantique. 130
3.19 Prédire les cas d'échec à partir de la représentation sémantique. Côté gauche : la légende et Q/A sont cohérents, mais pas assez riches pour prédire l'étiquette «girafe». Droite : la représentation sémantique est incorrecte, ce qui conduit à l'inférence d'étiquettes erronées. Dans de tels cas, le goulot d'étranglement peut être utilisé pour le débogage. 131
3.20 Interface utilisateur pour collecter les annotations de prédiction d'échec. L'image est présentée avec sa représentation sémantique générée (description + dialogue). Chaque étiquette peut être annotée avec l'un des 3 états : faux négatif, faux positif et correct. 132

INTRODUCTION

Le domaine de la vision par ordinateur a des applications dans de nombreux secteurs (l'aéronautique, les voitures autonomes, les télécommunications ...). Un des problèmes centraux est la reconnaissance visuelle dont le but est d'analyser automatiquement le contenu des images (objets, actions, scènes). Ce problème, qui peut sembler simple pour un humain, est très complexe pour la machine, l'image étant représentée par un ensemble de nombres sans signification sémantique explicite.

Pour développer des machines capables d'interpréter le contenu d'une image, la communauté de vision par ordinateur a proposé un nombre important de représentations visuelles. Durant de nombreuses années, la plupart des modèles de représentation d'images nécessitaient une expertise métier pour concevoir des extracteurs de caractéristiques performants [126, 144]. Récemment, les approches par apprentissage profond se sont imposées : leur particularité est d'apprendre l'extraction des caractéristiques conjointement à la tâche [66, 83, 142]. L'un des premiers succès

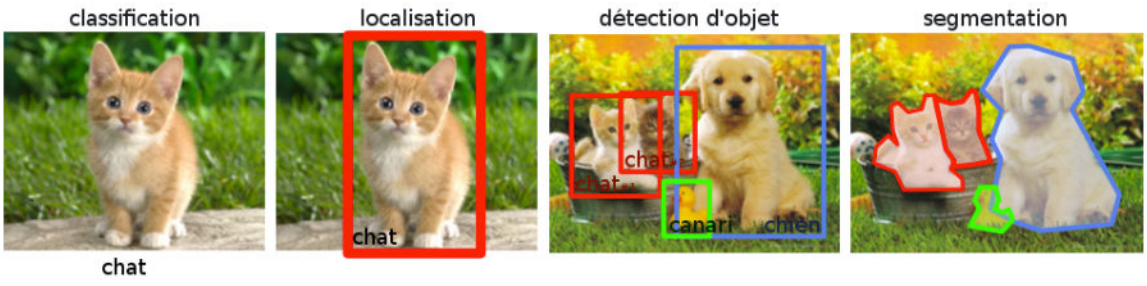


FIGURE 0.1 – *Image extraite de [4]* : les différentes tâches de vision par ordinateur.

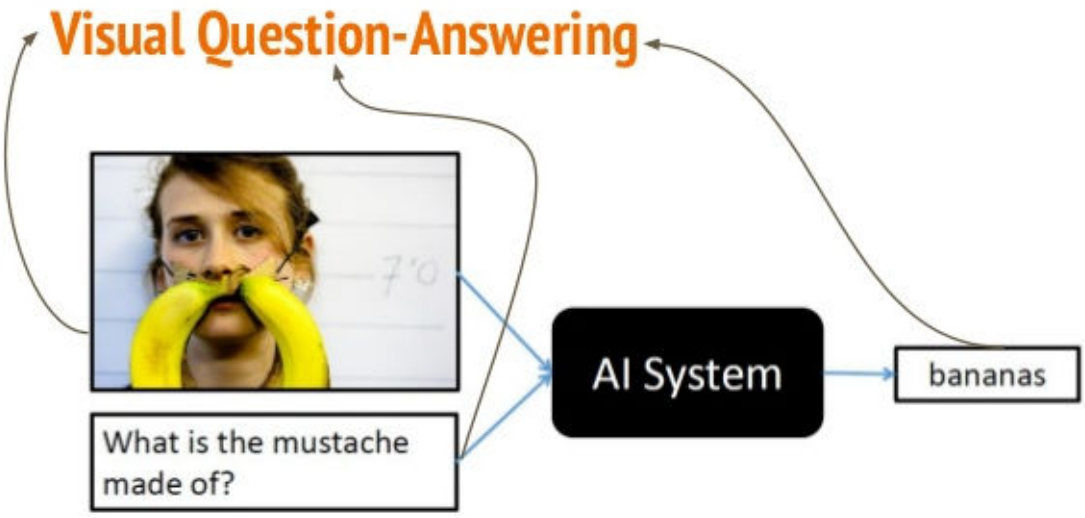


FIGURE 0.2 – *Image extraite de [9]* : l'objectif de la tâche consiste à répondre à une question posée sur une image, en langage naturel.

est la victoire d'un réseau convolutif au challenge ILSVRC en 2012 pour une tâche de classification d'image [83]. Cette victoire peut s'expliquer par deux raisons : une quantité de données étiquetées disponibles de plus en plus massives et l'utilisation de processeurs graphiques qui autorisent le calcul distribué à grande vitesse.

Suite à cela, l'apprentissage profond a été utilisé pour des tâches nécessitant une compréhension sémantique plus approfondie (voir figure 0.1), notamment en détection d'objets [113, 118, 142], qui combine la localisation ainsi que la classification des objets présents, en segmentation sémantique, où chaque pixel se voit assigner une classe et qui requiert une compréhension globale de la scène, utilisée par exemple pour de la classification d'images aériennes [13] ou dans le domaine médical [76].

Plus récemment, des tâches impliquant du «sens commun» ont émergé dans la littérature. La tâche appelée «Visual Question Answering» [9], consiste à répondre à une question posée sur une image, en langage naturel (voir figure 0.2). Le modèle est confronté à plusieurs types de questions concernant différentes caractéristiques d'objet : couleurs, formes, positions, relations... La tâche de dialogue visuel [37] quant à elle, se

présente sous la forme d'un échange en langage naturel entre l'utilisateur et le modèle à propos d'une image. La notion de continuité entre les questions et les réponses échangées rend obligatoire l'utilisation de modèle profond à mémoire.

Contexte

Limites de mise en pratique des modèles Malgré les très bons gains en performance des modèles profonds, leur mise en pratique reste limitée.

L'approche standard en apprentissage automatique présuppose l'existence d'un ensemble de données volumineux pour optimiser les paramètres du modèle. Des efforts continus ont été déployés pour collecter des corpus d'images plus larges en nombre de données et en couverture de classe. Cependant, la création d'annotations fines d'images et de haute qualité est difficile, coûteuse et longue. Pour les classes génériques telles que «voiture» ou «oiseau», les données sont abondantes, contrairement aux classes plus «fines» telles que le «guiraca bleu». De plus, à mesure que de nouvelles entités visuelles apparaissent, les annotations doivent être réactualisées et les classifieurs réappris. C'est pourquoi dans ce manuscrit nous proposons deux approches permettant la reconnaissance d'objet sans exemple de référence, le problème est connu dans la littérature sous le nom de classification zero-shot, l'introduction d'une description sémantique permettant de se substituer en données d'apprentissage.

Une seconde limitation de ces approches est qu'il est difficile pour l'utilisateur d'interpréter le résultat retourné, car obtenu après de nombreuses et complexes transformations. Cette opacité limite leur mise en pratique dans des domaines critiques, tels que la conduite autonome ou le diagnostic médical, où des éléments explicatifs sont indispensables pour prouver la sécurité des étapes de décision. Le travail présenté dans le chapitre 3 tente de répondre à ce problème d'interprétation à l'aide d'une représentation intermédiaire purement textuelle, autorisant alors la détection des défaillances de prédiction.

Rareté des données de référence

Le problème de la rareté des données d'apprentissage a été abordé de différentes manières dans la littérature.

Apprentissage faiblement supervisé L'apprentissage faiblement supervisé est un terme générique couvrant une variété d'études qui tentent de construire des modèles prédictifs à partir de données partiellement annotées.

Cette supervision faible implique un apprentissage initial (supervisé) avec un petit ensemble de données étiquetées. Le reste des données, généralement non labellisées, sont utilisées pour affiner le modèle ou incorporées comme données d'entraînement après la prédiction de leur étiquette par un «oracle» (non supervisé). Formellement, l'objectif est l'optimisation de la fonction $f : \mathscr{X} \mapsto \mathscr{Y}$ depuis un ensemble de données $D = \{(x_1, y_1), ..., (x_m, y_m), x_{m+1}, ..., x_n\}$ où m correspond au nombre d'exemples annotés et $p = n - m$ le nombre de données non annotées en général en plus grand volume ($m << p$).

On distingue deux approches, à savoir l'apprentissage actif et l'apprentissage semi-supervisé. L'apprentissage actif suppose qu'il existe un «oracle», tel qu'un expert humain, qui peut être interrogé pour obtenir des étiquettes de vérité terrain pour des instances non étiquetées. En revanche, l'apprentissage semi-supervisé tente d'exploiter automatiquement les données non étiquetées en plus des données étiquetées pour améliorer les performances d'apprentissage, aucune intervention humaine n'est supposée, l'étiquette est alors inférée à partir du modèle.

Une telle situation se produit pour diverses tâches, par exemple, en catégorisation d'images [45, 147], les étiquettes de vérité de terrain sont données par des annotateurs humains ; il est facile d'obtenir un grand nombre d'images sur Internet, alors que seul un petit sous-ensemble d'images peut être annoté en raison du coût humain. L'apprentissage

faiblement supervisé est également utilisé pour des tâches de détection d'objet [12], d'apprentissage de caractéristique [75] ou encore en segmentation sémantique [151].

Adaptation de domaine L'apprentissage par transfert est la ré-utilisation d'un réseau pré-entraîné sur un problème pour l'appliquer sur une tâche différente. Cette technique est populaire en apprentissage profond, car elle permet l'optimisation d'un réseau à de nouvelles problématiques en utilisant des ensembles de données relativement petits. Ceci est très avantageux, car pour la plupart des problèmes du monde réel, les jeux de données d'entraînement massifs étiquetés ne sont pas disponibles. De plus, les réseaux convolutifs entraînés sur un large jeu de données fournissent des représentations d'image très performantes [120].

L'adaptation de domaine est un cas particulier des techniques d'apprentissage par transfert. Ce scénario survient lorsque nous cherchons à apprendre à partir d'une distribution de données source D_S, un modèle performant sur une distribution de données cible D_T différente (mais connexe). L'objectif est alors d'optimiser la fonction f (à partir d'échantillons étiquetés provenant de D_S) pour qu'elle étiquette au mieux de nouvelles données issues du domaine cible D_T.

La question majeure soulevée par ce problème est la suivante : si un modèle a été appris à partir d'un domaine source, quelle sera sa capacité à étiqueter correctement des données provenant du domaine cible ?

Ce type d'approche est très utile en classification visuelle lorsque des images ont des contenus sémantiques similaires (mêmes classes d'objets), mais ont été récoltées dans des conditions différentes [95, 134, 149]. Ces techniques sont également utilisées pour de nombreuses tâches [34], notamment en reconnaissance d'action [49], d'identification faciale [140] ou encore pour de l'estimation de pose 3D [165].

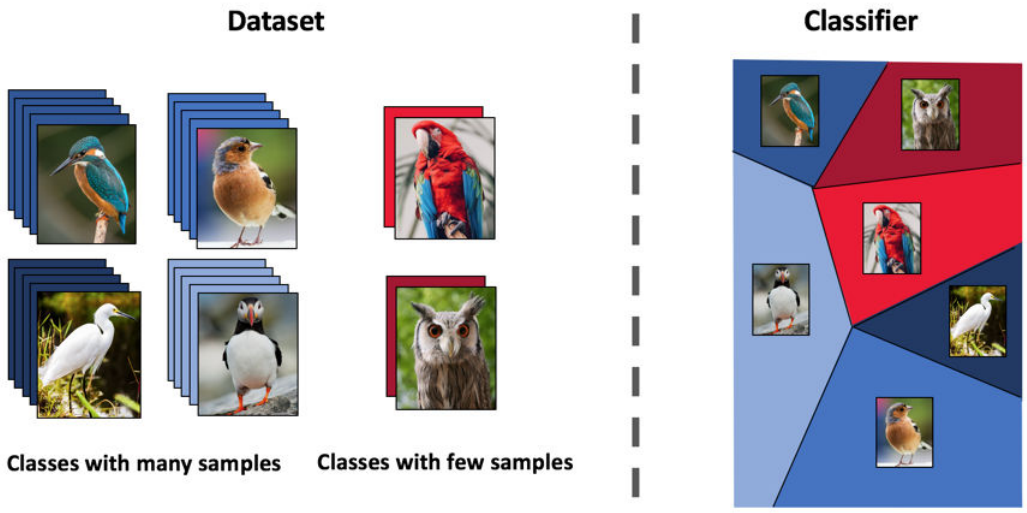


FIGURE 0.3 – *Image extraite de [1]* : les espèces d'oiseaux communes sont représentées par des images facilement disponibles, contrairement aux classes des espèces rares qui sont constituées de seulement quelques exemples.

Apprentissage avec un nombre réduit d'exemples La reconnaissance visuelle à partir d'un nombre réduit d'exemples fait référence à l'apprentissage d'algorithmes utilisant un très petit ensemble de données d'entraînement (une ou une dizaine d'images par classe). Les techniques d'optimisation standard ont tendance à sur-apprendre sur ces données.

L'idée est de transférer les connaissances apprises sur les catégories d'objets pour lesquelles de nombreuses données sont disponibles pour classer les nouvelles étiquettes (voir figure 0.3).

Apprentissage sans exemple Les approches mentionnées dans les paragraphes précédents sont tributaires d'un ensemble de données d'entraînement correctement étiquetées pour l'ensemble des classes à reconnaître. Cette restriction est difficilement satisfaite en pratique, c'est pourquoi un nouveau type d'approche est apparu dans la littérature, appelé classification «zero-shot».

Les approches zero-shot permettent d'introduire de nouveaux modèles

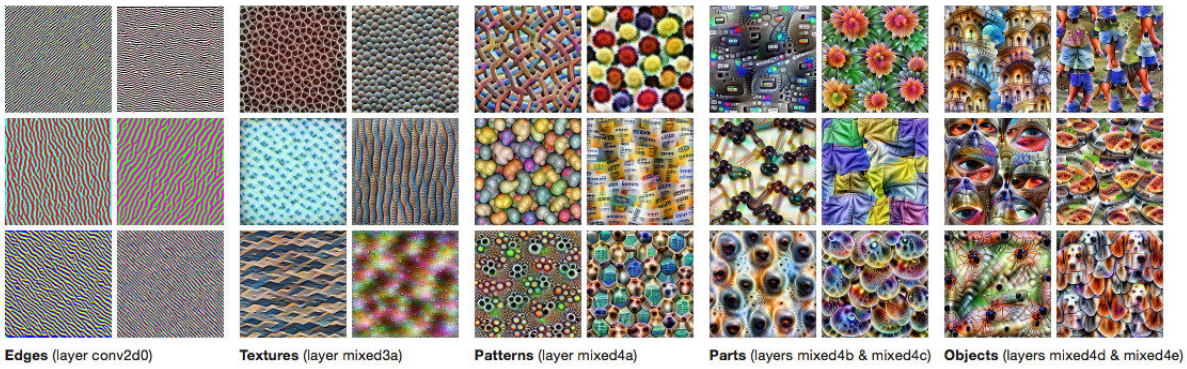


FIGURE 0.4 – *Image extraite de [5] :* visualisation des couches intermédiaires d'un réseau convolutif.

d'objets dans un système de reconnaissance visuelle à partir d'une description sémantique (attributs, texte, définition de dictionnaire...). Une représentation sémantique fournit une description de haut niveau d'une classe et permet d'établir un lien entre celle-ci et les différents concepts visuels connus. Dans les deux premiers chapitres de ce manuscrit, les modèles reposent sur l'existence d'un ensemble d'apprentissage étiqueté de classes vues et sur la connaissance de la façon dont chaque classe non vue (sans exemple de référence) est sémantiquement liée aux classes vues.

Intelligibilité de la décision

La décision prise par les modèles profonds, de par la complexité des calculs, est difficilement explicable. Ceci limite leur mise en pratique dans de nombreux domaines critiques où la compréhension de la prise de décision est décisive.

Pour répondre à cette problématique, deux axes de recherche ont émergé :

- Visualisation des couches intermédiaires [42, 101, 119, 169] : fournit une première intuition sur le fonctionnement des modèles, mais sans apporter une explication précise (voir figure 0.4).
- Génération d'explications [68, 112, 127, 136] : soit proposées sous

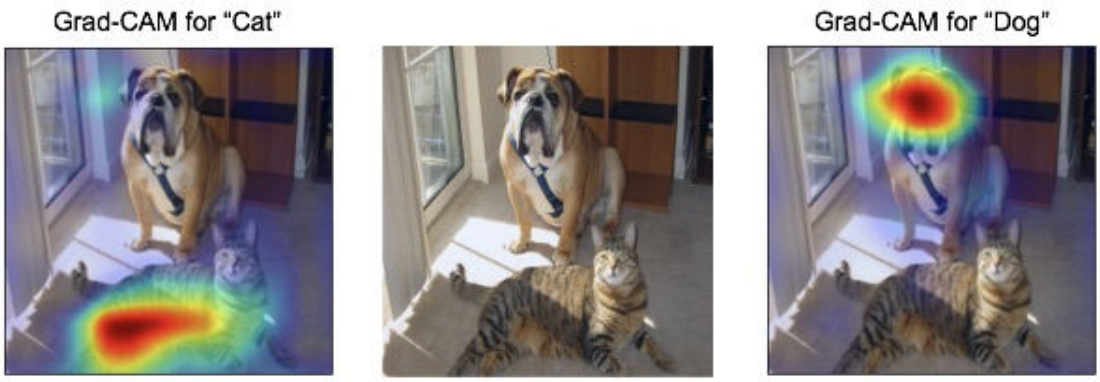


FIGURE 0.5 – *Image extraite de [136] :* les cartes de saillance générées permettent de visualiser les zones de l'image utilisées pour la prise de décision.

la forme d'une représentation visuelle dans l'espace d'entrée (cartes de chaleur ou des cartes de saillance) (voir figure 0.5) ou soit comme descriptions textuelles

Le travail présenté dans le chapitre 3 fusionne ces deux axes de recherche. Nous proposons un goulot d'étranglement sémantique qui joue le rôle d'une explication du processus de prédiction. L'aspect intelligible de la représentation offre l'opportunité d'examiner sur quelles informations les prédictions sont faites, et de décider éventuellement de les rejeter.

Organisation de ce manuscrit

Ce manuscrit est organisé autour de trois chapitres : les deux premiers traitent de la reconnaissance visuelle sans exemple d'apprentissage, le dernier s'intéresse à l'intelligibilité d'une décision par un modèle statistique profond.

Thème 1 : Reconnaissance visuelle sans exemple d'apprentissage

L'objectif de ces chapitres est de développer des modèles statistiques permettant la reconnaissance d'objets sans exemple de référence (classification zero-shot).

- Chapitre 1 - *Apprentissage d'un espace sémantique optimal pour la reconnaissance visuelle sans exemple d'apprentissage :* ayant pour objectif d'obtenir une représentation sémantique plus performante, nous contrôlons la structure de l'espace grâce à l'apprentissage d'une métrique, capable à la fois de sélectionner et de transformer la distribution des données originales.
- Chapitre 2 - *Génération de caractéristiques visuelles conditionnées par représentation sémantique :* nous proposons de générer des caractéristiques visuelles à partir d'une représentation sémantique. Une fois générées, elles sont utilisées pour l'apprentissage d'un classifieur discriminant.

Thème 2 : Intelligibilité du processus de décision

Dans ce chapitre nous abordons la question de l'intelligibilité des calculs pour les tâches de vision par ordinateur.

- Chapitre 3 - *Goulot d'étranglement sémantique pour la détection de défaillance :* Notre proposition est d'introduire un goulot sémantique dans le processus de traitement, la représentation de l'image est alors exprimée entièrement en langage naturel. Nous montrons

que notre approche est capable de détecter des défaillances de prédiction.

CHAPITRE 1

APPRENTISSAGE D'UN ESPACE SÉMANTIQUE OPTIMAL POUR LA RECONNAISSANCE VISUELLE SANS EXEMPLE D'APPRENTISSAGE

L'objectif de ce chapitre est de développer un modèle d'apprentissage statistique permettant la reconnaissance de classes d'objets sans exemple de référence (classification zero-shot).

Les algorithmes de zero-shot récents [6, 84] suivent un mécanisme de décision commun. Ils reposent sur le calcul d'une similarité ou d'une fonction de cohérence liant les descripteurs d'images et la description sémantique des classes. Ces liens sont donnés en apprenant deux plongements - le premier de la représentation d'image à l'espace sémantique et le second de l'espace de classe à l'espace sémantique - et en définissant une façon de décrire les contraintes entre l'espace de classe et l'espace visuelle, ces deux espaces étant fortement interdépendants.

Bien que ces approches répondent en partie à notre problématique, leurs performances de reconnaissances visuelles restent limitées.

Dans ce chapitre, nous suggérons qu'un meilleur contrôle de la structure de l'espace sémantique (dans notre cas un ensemble d'attributs) est

au moins aussi important que l'étape classification. L'idée fondamentale est d'obtenir une distribution d'attributs optimale grâce à l'apprentissage d'une métrique, capable à la fois de sélectionner et de transformer la distribution des données originales.

Pour ce faire, nous proposons une nouvelle méthode basée sur un critère de coût multi-objectifs :

1. le 1er objectif contrôle la qualité du plongement sémantique ;
2. le 2ème, agit sur l'espace sémantique grâce à une métrique de Mahalanobis.

Après avoir décrit notre approche, nous la validons sur 3 tâches. La première concerne la reconnaissance d'objets sans exemple de référence, appelée aussi «classification zero-shot». La deuxième autorise un faible nombre d'images d'entraînement par classe. Et la dernière se porte sur la tâche de recherche d'images dans une base de données.

1.1 Introduction

En vision par ordinateur, on désigne par reconnaissance d'objets une méthode consistant à attribuer une classe ou une catégorie à une image numérique. Ces méthodes font souvent appel à l'apprentissage statistique et ont des applications dans de multiples domaines, tels que la recherche d'images par le contenu, la vidéo surveillance, la reconnaissance de visages pour la photographie ou encore le traitement d'images aériennes.

Les algorithmes de reconnaissance visuelle d'objets sont en progrès constant. Leurs succès récents sont fondés sur une exploitation de techniques efficaces d'apprentissage statistique et surtout sur la disponibilité de bases d'images annotées de références massives (plusieurs centaines de milliers d'images) [39].

Les algorithmes dits d'apprentissage profond («deep learning») en particulier ont montré récemment un gain notable dans certaines compéti-

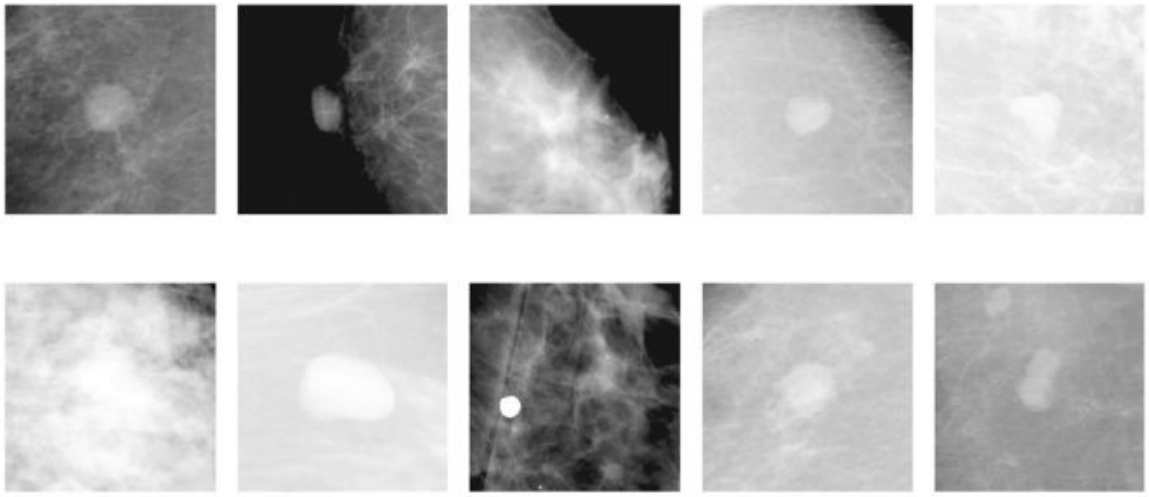


FIGURE 1.1 – *Image extraite de [81] :* données médicales provenant d'un système de mammographie. Détection automatique de la nature cancérigène du kyste. Ce type d'approche à un coût très élevé notamment dans l'acquisition de l'appareil ainsi qu'en temps et effort humain.

tions de détection ou reconnaissance d'objets dans des images [65, 83, 148].

Dans les situations réelles (voir figure 1.1), disposer de données massives et statistiquement pertinentes – elles sont souvent de contextes disparates, de qualités images variées, et de conditions d'observation hétérogènes – est une condition rarement satisfaite, et limite l'utilisation pratique des techniques d'apprentissage massif.

De plus l'ensemble des catégories possibles à distinguer est potentiellement illimité (voir figure 1.2) ce qui rend difficile la collecte et l'annotation de données, surtout pour les techniques d'apprentissage profond nécessitant de larges bases de données annotées. Par ailleurs, de nouvelles catégories sont susceptibles de devoir être reconnues, les approches standards de la littérature nécessitent alors le ré-apprentissage complet du modèle avec de nouvelles données annotées.

L'objectif de ce chapitre est la conception et l'évaluation d'une démarche permettant d'introduire de nouveaux modèles d'objets dans un

FIGURE 1.2 – *Image extraite de [3]* : l'ensemble des catégories possibles est potentiellement infini, ce qui rend difficile la collecte et l'annotation de données pour toutes les classes (de plus de nouvelles catégories émergent de manière ininterrompue).

système de reconnaissance visuelle à partir d'une description sémantique (attributs, texte, définition de dictionnaire...).

En classification d'objets, l'objectif est d'annoter une image avec son label correspondant, ce label décrit au mieux le contenu visuel. Cette tâche de prédiction peut être présentée sous la forme d'un problème d'apprentissage statistique : on cherche à optimiser depuis des données annotées $\mathscr{D} = \{\mathbf{x_i}, y_i\}_{i=1}^{N}$ une fonction $\mathscr{F} : \mathscr{X} \rightarrow \mathscr{Y}$ capable de prédire à partir d'une entrée $x \in \mathscr{X}$, l'espace visuel $\mathscr{X} \in \mathbb{R}^D$, une classe $y \in \mathscr{Y}$, l'espace label $\mathscr{Y} \in \mathbb{R}$.

Les approches d'apprentissage supervisées standards sont incapables de généraliser à de nouvelles classes, c'est à dire des images provenant de classes non présentes pendant l'apprentissage.

FIGURE 1.3 – *Image extraite de [2] Segway : véhicule électrique monoplace, constitué d'une plateforme, muni de deux roues parallèles et d'un manche de maintien et de conduite sur laquelle l'utilisateur se tient debout.* Inspirée par les capacités humaines, notre approche permet de reconnaître un objet à partir d'une représentation sémantique.

Dans ce chapitre, nous nous intéressons au cas où certaines classes ne sont pas représentées lors de la phase entraînement. Nous faisons alors la distinction entre deux types de classes, celles dites vues où des données de références sont disponibles lors de l'optimisation de la fonction de décision, et les classes non vues, sans donnée d'apprentissage, mais représentées lors de l'étape d'évaluation. Les ensembles de classes vues $\mathcal{Y}_s$ et non vues $\mathcal{Y}_u$ sont disjoints $\mathcal{Y}_s \cap \mathcal{Y}_u = \varnothing$.

Une approche possible pour autoriser une meilleure prise en compte de l'hétérogénéité des données d'apprentissage est de passer par une représentation intermédiaire d'attributs visuels d'objets calculés sur les images. Ces attributs sont chargés d'exprimer, dans un vocabulaire plus ou moins riche, soit des composants des objets ou actions (roue, phare, œil, menton, bras droit levé, mouvement circulaire de la main...) soit des configurations globales (allongé, profil, break...) soit des caractéristiques intensives (couleur) ou extensives (longueur, durée).

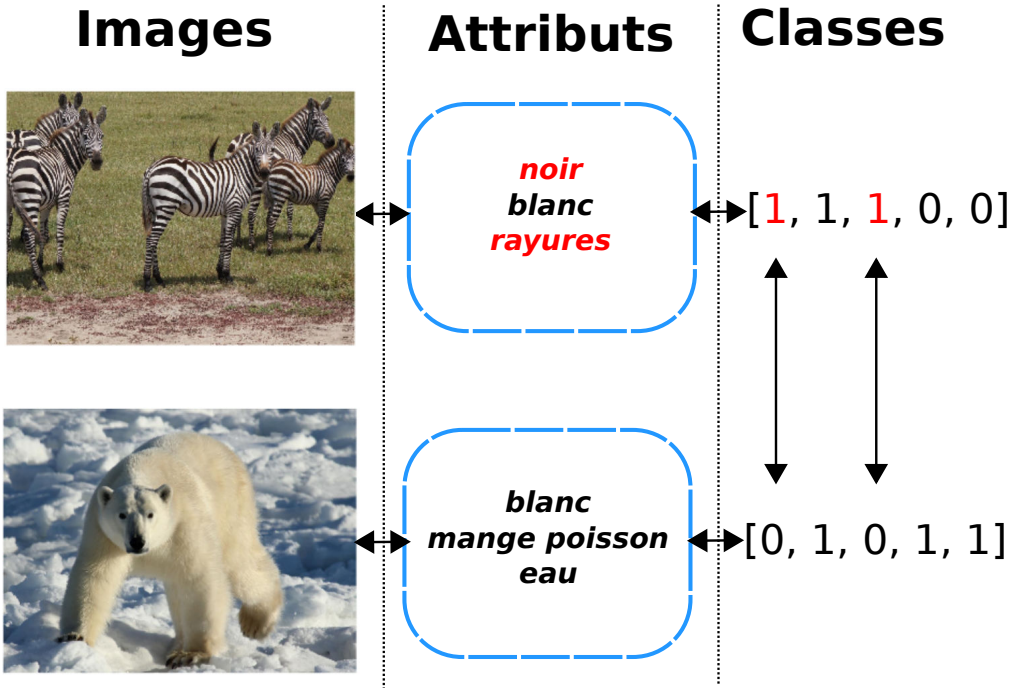


FIGURE 1.4 – L'utilisation d'attributs permet de construire un nouvel espace de représentation pour l'apprentissage. Chaque classe est représentée par un vecteur d'attributs correspondant à la présence ou l'absence de ceux-ci. Elle permet de définir les modèles par description, par opposition à une modélisation à partir d'un ensemble d'exemples, et rend possible la modélisation sans donnée de référence («zero-shot learning»).

Une analogie simple peut être faite avec la capacité humaine d'imaginer un objet grâce à une description textuelle (voir figure 1.3). La description, si assez discriminante, permet la reconnaissance de l'objet sans exemple visuel préalable.

L'utilisation d'attributs pour modéliser des classes permet de définir les modèles par description, par opposition à une modélisation à partir d'un ensemble d'exemples, et rend possible la modélisation sans donnée de référence. L'attribut peut être détecté par la machine et compris par l'humain. Chaque classe est alors représentée par un vecteur d'attributs (voir figure 1.4) correspondant à la présence ou l'absence de ceux-ci pour un objet donné. L'ensemble de données d'apprentissage est composé d'une nouvelle information, les attributs $\mathbf{a_i} \in \mathscr{A}$ présents pour chaque classe ou

chaque image $\mathscr{D} = \{\mathbf{x_i}, \mathbf{a_i}, y_i\}_{i=1}^{N}$.

Dans ce chapitre, nous suggérons de mieux contrôler la structure de l'espace d'attribut. L'idée fondamentale est d'améliorer la distribution des attributs en apprenant une métrique capable à la fois de sélectionner et de transformer la distribution de données d'origine. Nous validons empiriquement l'idée que l'optimisation conjointe du plongement d'attributs et de la métrique de classification, dans un cadre multi objectif, permet de meilleures performances sur deux tâches visuelles : la reconnaissance et la recherche d'objet dans une base de données.

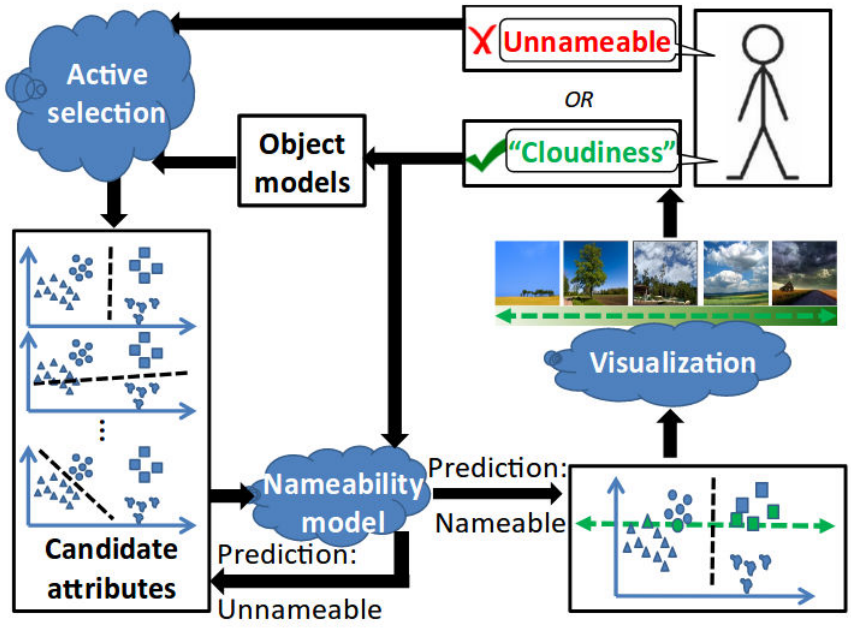


FIGURE 1.5 – *Figure extraite de [121] :* introduit une approche pour définir un vocabulaire d'attributs à la fois humainement compréhensible et discriminant. Le système prend en entrée des images étiquetées par objet/scène et renvoie en sortie un ensemble d'attributs obtenus par interactions avec un annotateur humain, chargé de distinguer les bons aux mauvais attributs.

1.2 État de l'art

1.2.1 Création d'un ensemble d'attributs

Les méthodes d'apprentissage zero-shot reposent sur l'utilisation de représentations intermédiaires, généralement un ensemble d'attributs. Ce terme peut toutefois englober différents concepts. Pour Lampert [84], ils indiquent la présence ou l'absence d'une propriété d'objet, en supposant que les attributs sont des propriétés nommables (couleur ou présence ou absence d'une certaine partie, etc.). L'avantage des attributs est qu'ils peuvent être utilisés facilement pour définir de nouvelles classes.

Trouver un ensemble discriminant et significatif d'attributs peut parfois être difficile. [44, 121] aborde ce problème en proposant une approche

interactive qui découvre des attributs à la fois discriminants et sémantiquement significatifs, en utilisant un système de recommandation via interactions humaines (voir figure 1.5). Une alternative pour identifier des attributs sans étiquetage humain, est d'extraire la description textuelle qui accompagne généralement les images échantillonnées à partir d'internet, telle que proposée par [19]. Les attributs peuvent également être structurés de façon hiérarchique [7, 131, 150] ou être extraits à partir de descriptions textuelles [7, 14, 47].

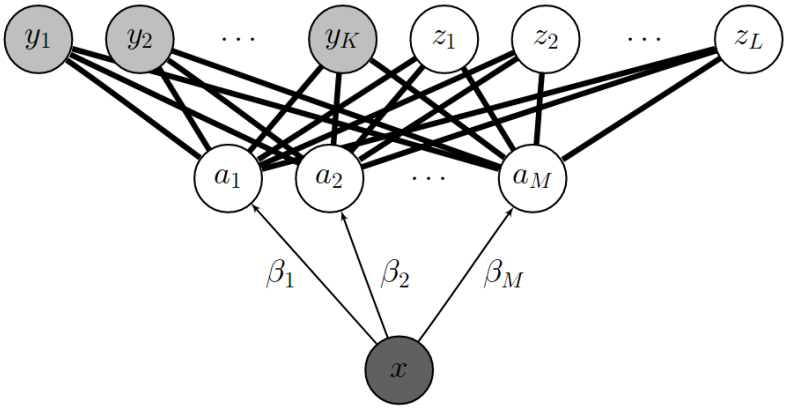
Un autre inconvénient des attributs définis par les humains est qu'ils peuvent être redondants ou non adaptés à une tâche de classification. Yu [168] propose de générer automatiquement des attributs discriminants pour chaque catégorie et de les utiliser comme un moyen de transférer la connaissance inter catégorie.

À côté de ces articles qui considèrent les attributs comme sémantiquement pertinents pour les humains. Certains auteurs désignent par «attributs» tout espace latent fournissant une représentation intermédiaire entre l'image et la classe et qui peut être utilisé pour transférer de l'information à des classes non vues. C'est typiquement le cas de [156] qui détecte les attributs difficilement détectables et redondants, et les convertis en attributs discriminants latents.

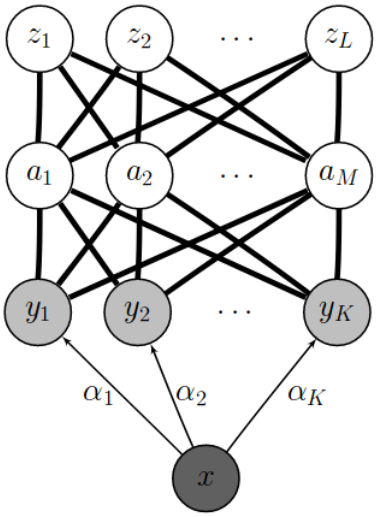
Cependant les techniques mentionnées limitent le concept d'attribut à la présence ou l'absence d'une propriété et ne permettent pas des relations sémantiques plus «générales». Pour contrer cette limitation, [122] propose de modéliser de façon relative les attributs ((b) sourit plus que (a), mais moins que (c)), via l'apprentissage d'une fonction de rang.

1.2.2 Utilisation d'attributs comme représentation intermédiaire

Les méthodes «Direct Attribute Prediction» (DAP) et «Indirect Attribute Prediction» (IAP), proposées par [84], sont l'une des premières



(a) Direct Attribute Prediction (DAP)



(b) Indirect Attribute Prediction (IAP)

FIGURE 1.6 – *Figure extraite de [84]* : (a) par le biais de détecteurs d'attributs, le modèle extrait la représentation de l'image **x**. Chaque label d'entraînement y_i est représenté par la présence ou l'absence d'attributs a_i. Les relations classe-attribut sont fixes (lignes épaisses). Au moment du test, les attributs prédits à partir de l'image permettent d'inférer la classe, même pour les classes non vues. (b) Les attributs **a** forment une couche de connexion intermédiaire entre les classes vues y et non vues z. Les attributs ne sont plus prédits à partir de l'image, mais des vraisemblances de celle-ci d'appartenir aux classes vues.

propositions de la littérature pour la tâche de classification zero-shot. L'idée générale est d'utiliser un ensemble d'attributs comme un espace intermédiaire entre les images et la couche d'étiquettes (voir figure 1.6). Pour l'approche DAP, le plongement sémantique de l'image est réalisé par des détecteurs d'attributs indépendants, la similarité entre deux représentations sémantiques (une prédite à partir de la représentation d'image, une par la classe) est donnée comme la probabilité de connaître la classe sachant les attributs détectés. Pour IAP, les attributs forment une couche de connexion intermédiaire entre les classes vues et non vues. Les attributs ne sont plus prédits à partir de l'image, mais des vraisemblances de celle-ci d'appartenir aux classes vues.

Ces deux approches souffrent de plusieurs lacunes. Premièrement, l'apprentissage se fait en deux étapes, les détecteurs d'attributs sont appris indépendamment du classifieur. Les performances peuvent alors être optimales pour la détection d'attribut, mais pas nécessairement pour la tâche finale. Aussi, les approches proposées se focalisent uniquement sur un espace intermédiaire d'attributs, qui sont une source coûteuse à obtenir ; avec un étiquetage humain en outre pas toujours fiable. Une autre limitation de ces méthodes est qu'elles ne permettent pas d'ajouter de façon incrémentale de nouvelles données, si par exemple des images venaient à être disponibles pour les classes non vues. Enfin du fait de l'indépendance des détecteurs, la corrélation entre les attributs n'est pas prise en compte, la détection d'un attribut pourrait pourtant permettre de confirmer ou d'infirmer la présence d'un autre attribut.

1.2.3 Plongement de l'image et de son l'étiquette dans un espace sémantique commun

Inspiré par [51], [6] et [132] proposent une alternative permettant de remédier aux limitations de DAP/IAP, par l'intermédiaire d'un plongement linéaire entre l'image et son étiquette.

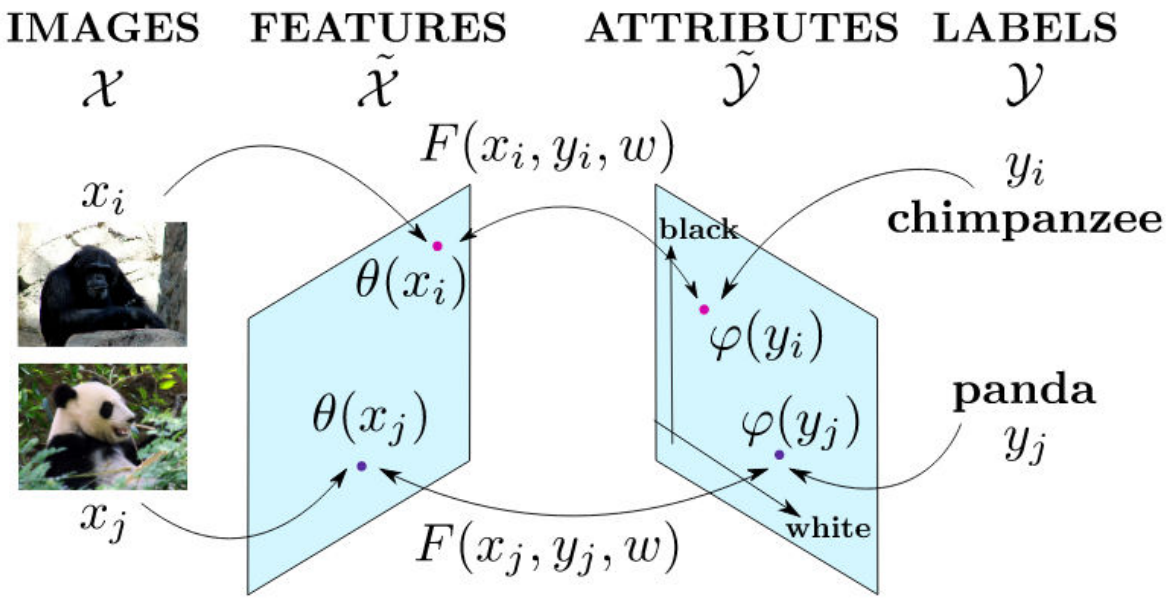


FIGURE 1.7 – *Image extraite de [6]* : $\theta(\mathbf{x})$ correspond aux caractéristiques de l'image $\mathbf{x}$. Les étiquettes de classes y sont représentées par un ensemble d'attributs $\phi(y)$. La fonction de score est $F(\mathbf{x}, y; \mathbf{W})$.

Pour cela, [6] introduit une fonction mesurant la cohérence entre une image et son étiquette, les paramètres de cette fonction sont appris pour s'assurer que, étant donné une image, la classe correcte obtient le score le plus élevé (voir figure 1.7). Cette fonction de cohérence a la forme d'une relation bilinéaire $\mathbf{W}$ associant les caractéristiques visuelles $\theta(\mathbf{x})$ et la représentation de l'étiquette $\phi(y)$ comme $F(\mathbf{x}, y; \mathbf{W}) = \theta(\mathbf{x})^t \mathbf{W} \phi(y)$. [162] propose de représenter la fonction de plongement comme un ensemble de relations bilinéaires entre l'image et son étiquette. Quant aux travaux de [132], la forme spécifique de régularisation proposée permet de résoudre analytiquement l'optimisation des paramètres $\mathbf{W}$.

À la différence de nos travaux, aucun des articles cités ([6], [132], [162]) ne contrôle la structure statistique de l'espace des attributs.

D'autres formes de similarité ont été proposées dans la littérature, notamment [63] qui introduit l'idée de similarité ordinale entre classes (par exemple $S(chat, chien) > S(chat, camion)$). Ils affirment que non

seulement ce type de similarité peut être suffisante pour distinguer un chat d'un camion, mais aussi que cela semble une représentation plus naturelle puisque la similarité ordinale est invariante à l'échelle et à toute transformation monotone.

Il convient également de mentionner le travail de [74] qui tire parti des statistiques d'erreur de prédiction des attributs, pour former des modèles zero-shot sous forme de forêt aléatoire. Mais aussi de [161], qui après avoir généré les représentations sémantiques des classes à partir des descriptions d'image disponibles sur internet, mesure leur similitude avec les images en les projetant dans un espace commun de grande dimension. [117] représente les images par leurs vraisemblances d'appartenir aux classes vues. Cela fournit une représentation pour chaque image, qui est ensuite utilisée pour extrapoler l'appartenance à une des classes non vues.

Tout comme proposé dans ce chapitre, [108] utilise une métrique dans un espace sémantique. Cette métrique est optimisée dans un contexte de classification d'image pour un modèle de k plus proches voisins. Cependant contrairement à notre approche, leurs travaux ne considèrent pas la problématique de classification zero-shot, mais celle appelée «one shot learning» (détection visuelle à partir d'une seule image).

1.2.4 Reconnaissance visuelle avec un nombre réduit d'exemples

La reconnaissance visuelle avec un nombre réduit de données («few-shot learning») fait référence à la compréhension de nouveaux concepts à partir de quelques exemples seulement.

Basé sur l'idée que si il n'y en a pas suffisamment d'instances disponibles pour apprendre les paramètres du modèle, il est alors nécessaire d'en créer de nouvelle. [43] exploite un ensemble de sources de données externes et les incorpore par le biais d'un apprentissage semi-supervisé. Les nouvelles images sont sélectionnées suivant une mesure de similarité

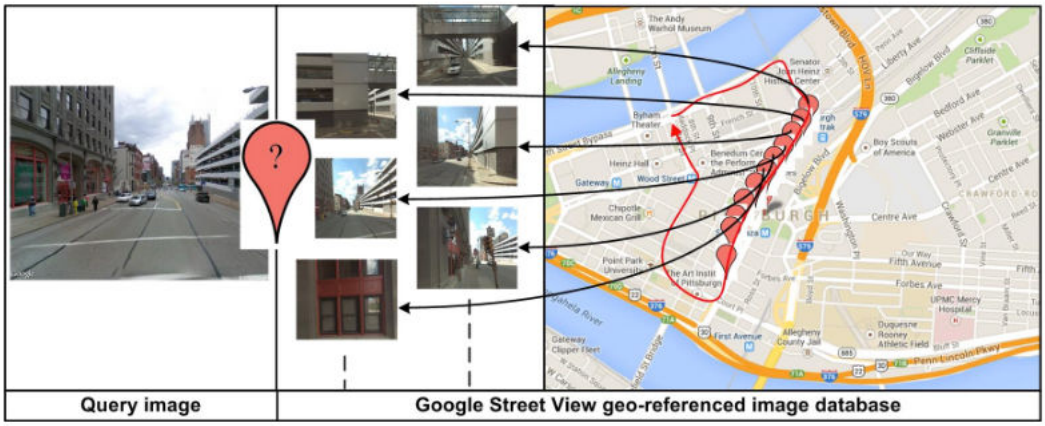


FIGURE 1.8 – *Image extraite de [86] :* localise un lieu à partir d'une image, ainsi qu'à partir de ses transformations géométriques et photométriques.

avec les images étiquetées. Par la suite, la propagation des étiquettes est effectuée par le modèle et les images sont ajoutées à l'ensemble d'apprentissage, ce qui élargit le corpus. Une autre approche possible [105], consiste à générer de nouvelles données à l'aide d'un modèle génératif («generative adversarial network»). Après apprentissage de la distribution des données d'entraînements, le modèle permet d'échantillonner de nouveaux exemples pour les classes faiblement représentées.

Pour réduire le nombre d'itérations nécessaires à un algorithme d'optimisation standard, telle que la descente de gradient stochastique, [129] propose un méta-algorithme capable de cibler le sous-espace des paramètres autorisant la reconnaissance visuelle même avec un nombre réduit d'exemples. Enfin, pour éviter le sur apprentissage, [167] réduit la dimension de l'espace de recherche, en maximisant la similarité intra-classe et la diversité inter-classe des activations des paramètres du modèle.

1.2.5 Apprentissage de métrique et sélection de paires optimales

Le problème de l'apprentissage de métrique concerne l'optimisation d'une fonction de distance adaptée à une tâche particulière. Elle s'avère

utile lorsqu'elle est utilisée conjointement avec les méthodes de plus proches voisins ou d'autres techniques reposant sur la notion de distance ou de similitude.

L'objectif est d'apprendre une fonction de compatibilité qui mesure à quel point deux objets sont similaires. Elle est obtenue grâce à une collection de données organisée sous la forme de paires de points similaires et dissimilaires, les exemples d'une même classe se voient rapprochés, tout en séparant les données provenant des classes différentes.

Des travaux antérieurs [64, 66, 67] ont montré que des métriques conçues de manière appropriée peuvent grandement améliorer la précision de la classification d'un algorithme de plus proches voisins par rapport à la distance euclidienne standard.

Cela a motivé un grand nombre de travaux en vision par ordinateur, tels que les systèmes de recommandation visuelle [27, 104], la reconnaissance faciale [20, 33, 115] ou encore la localisation d'un lieu à partir d'une image [85, 86] (voir figure 1.8). Il a également été montré [71, 159] que la qualité de recherche des systèmes CBIR (Content-Based Image Retrieval) dépend fortement du critère utilisé pour définir la similarité entre les images.

Dans un problème d'apprentissage de métrique, alors que l'ensemble des paires positives est fixé et donné par les données d'apprentissage, les paires négatives peuvent être choisies plus librement.

Les techniques appelées «hard negative mining» consistent à sélectionner les paires négatives les plus informatives. Très utilisées en détection d'objet, elles permettent de sélectionner les fenêtres les plus incertaines à incorporer à l'ensemble d'entraînement [28, 142]. Cette stratégie est également utilisée en recherche de vidéo [166] pour la sélection de «frame».

Comme nous verrons dans la suite de ce chapitre, la sélection intelligente de paires négatives permet d'améliorer la robustesse des modèles aux exemples incertains et de diminuer le nombre d'itérations d'apprentissage.

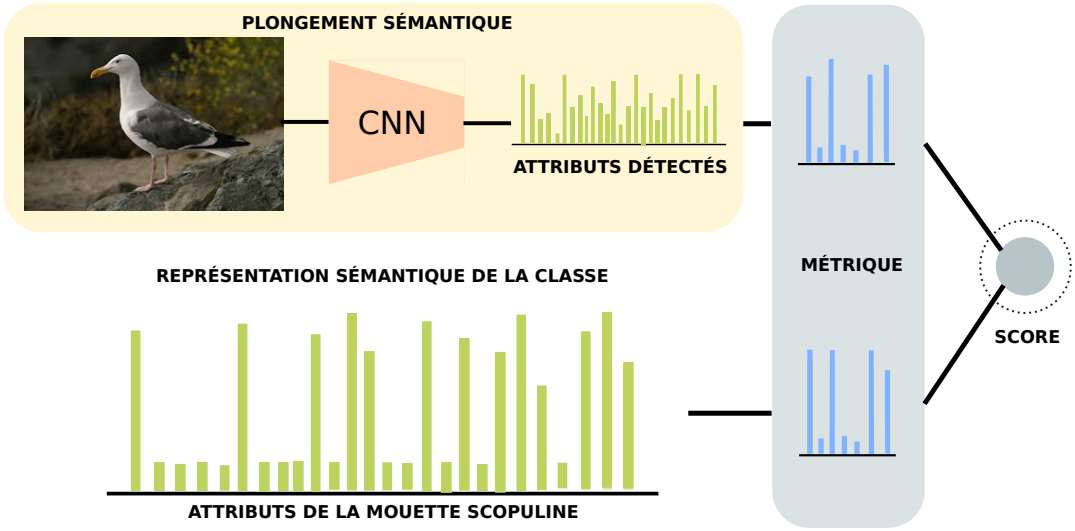


FIGURE 1.9 – À l'aide d'un réseau convolutif, l'image est plongée dans l'espace sémantique (attributs verts). Certains attributs sont difficiles à détecter, redondants ou encore inutiles. La métrique appliquée à l'image et à la représentation de la classe, est capable à la fois de sélectionner et de transformer la distribution de données originales (attributs bleus).

1.3 Apprentissage de métrique pour un espace d'attribut optimal

Dans cette section, nous suggérons qu'un meilleur contrôle de la structure de l'espace d'attribut est aussi important que l'étape d'inférence de classification (voir figure 1.9). L'idée fondamentale est d'obtenir une distribution d'attributs optimaux en apprenant une métrique capable à la fois de sélectionner et de transformer la distribution de données originales selon des critères de tâche zero-shot.

1.3.1 Classification zero-shot

Comme motivé dans l'introduction, nous abordons dans ce chapitre le problème de l'apprentissage d'un classifieur capable de discriminer entre un ensemble de classes où les données d'apprentissage ne sont disponibles

que pour un sous-ensemble, les classes dites *vues*.

L'ensemble de données d'apprentissage $\mathscr{D}_s$ est défini par une série de triplets $\{\mathbf{x}_\mathbf{i}^\mathbf{s}, \mathbf{a}_\mathbf{i}^\mathbf{s}, y_i^s\}_{i=1}^{N_s}$ où $\mathbf{x}_\mathbf{i}^\mathbf{s} \in \mathscr{X}$ est la donnée brute (image ou caractéristiques visuelles), $y_i^s \in \mathscr{Y}_s$ est l'étiquette de classe associée et $\mathbf{a}_\mathbf{i}^\mathbf{s}$ est une représentation sémantique de la classe (attributs, vecteur de mots ou texte) appartenant à $\mathscr{A}_s$.

Cette représentation sémantique doit :

1. contenir suffisamment d'informations pour discriminer l'ensemble des classes
2. être prédictible à partir des données brutes
3. déduire sans ambiguïté l'étiquette de la classe $y = l(\mathbf{a})$.

Dans un problème de classification zero-shot, tout ce que l'on connaît du nouveau domaine cible est l'ensemble des représentations sémantiques $\mathscr{A}_u$ des classes *non vues*. Le but est d'utiliser cette information et la structure de l'espace de représentation sémantique pour concevoir une fonction de classification f capable de prédire l'étiquette de classe $\hat{y} = f(\mathbf{x}; \mathscr{A}_u, \mathscr{D}_s)$. La fonction de classification f est généralement paramétrique et optimisée à partir d'un critère d'apprentissage empirique.

1.3.2 Score de compatibilité

Notre problème d'inférence peut être présenté sous la forme :

$$y^* = \arg\min_{y \in \mathscr{Y}} S(\mathbf{x}, y) \tag{1.1}$$

où $\mathbf{x} \in \mathscr{X}$ est un vecteur à valeurs réelles de caractéristiques images extraites d'une réseau convolutif. $y \in \mathscr{Y}$ est une autre modalité, dans notre cas une étiquette de classe. S est une mesure capable de quantifier la cohérence entre les deux modalités et permet d'obtenir la sortie y^*, l'association la plus cohérente. Dans cette formulation, plus le score est

petit, plus les échantillons sont cohérents. On peut considérer ce score comme une probabilité inversée ou une similarité hétérogène.

En essayant de concevoir un tel score de cohérence, l'un des aspects difficiles est de relier de manière significative les deux modalités. Une approche standard consiste à les incorporer dans un espace de représentation commun $\mathscr{A}$ où leur nature hétérogène peut être comparée. Cet espace peut être abstrait, c'est-à-dire que sa structure peut être obtenue à partir d'un processus d'optimisation, ou sémantiquement interprétable, par exemple une liste fixe d'attributs.

Soit $A_X(\mathbf{x})$ et $A_Y(y)$ les deux fonctions de plongement pour les modalités $\mathbf{x}$ et y, prenant leurs valeurs dans $\mathscr{X}$ et $\mathscr{Y}$ et produisant des sorties dans $\mathscr{A}$.

Dans ce travail, nous proposons de définir le score de cohérence comme une métrique sur l'espace de plongement commun $\mathscr{A}$. Plus précisément, nous utilisons l'approche de Mahalanobis comme métrique paramétrée par $\mathbf{W}_A$:

$$d_A(\mathbf{a}_1, \mathbf{a}_2) = \left\| (\mathbf{a}_1 - \mathbf{a}_2)^T \mathbf{W}_A \right\|_2, \tag{1.2}$$

En supposant que l'espace de plongement est un espace vectoriel, le score de cohérence est alors défini par :

$$S(\mathbf{x}, y) = d_A(A_X(\mathbf{x}), A_Y(y)) = \left\| (A_X(\mathbf{x}) - A_Y(y))^T \mathbf{W}_A \right\|_2. \tag{1.3}$$

La matrice de Mahalanobis $\mathbf{W}_A$ peut être interprétée comme une transformation linéaire de l'espace d'attribut vectoriel p-dimensionnel en un nouvel espace abstrait m-dimensionnel ($m << p$) optimal pour la tâche cible.

On s'attend à ce que la métrique améliore empiriquement la fiabilité du score de cohérence (équation 1.3) en choisissant la transformation linéaire appropriée.

Il nous reste maintenant à répondre à deux questions : comment définir le plongement sémantique ? ; Comment optimiser la transformation de

Mahalanobis ? Nous verrons par la suite que ces deux questions peuvent être résolues conjointement en optimisant un critère unique.

1.3.3 Plongement dans un espace d'attributs

Le principal problème abordé dans ce travail est de pouvoir discriminer une série de nouvelles hypothèses (ex. nouvelles classes) qui ne peuvent être spécifiées qu'en utilisant une seule modalité, le y avec notre notation.

Dans de nombreuses études de la littérature zero-shot, cette modalité est exprimée comme la présence ou l'absence d'un ensemble d'attributs.

L'espace de plongement le plus simple auquel on puisse penser est précisément cet espace d'attribut. Dans la suite de ce chapitre, on note $A_Y(y) = \mathbf{y}$, où $\mathbf{y}$ est la représentation d'attributs de la classe y définie au préalable par un expert.

Dans ce cas, le score de cohérence se simplifie comme suit :

$$S(\mathbf{x}, \mathbf{y}) = \left\| (A_X(\mathbf{x}) - \mathbf{y})^T \mathbf{W}_A \right\|_2 \tag{1.4}$$

L'étape suivante consiste à plonger directement la modalité $\mathbf{x}$ dans $\mathbf{y}$. Nous suggérons d'utiliser un plongement linéaire simple avec la matrice $\mathbf{W}_X$ et le biais $\mathbf{b}_X$, en supposant que $\mathbf{x}$ appartienne à un espace vectoriel d-dimensionnel. Dans notre cas un modèle plus complexe est difficilement exploitable en raison du faible nombre d'exemples d'apprentissage.

Cela peut être exprimé par :

$$A_X(\mathbf{x}) = \max(0, \mathbf{x}^T \mathbf{W}_X + \mathbf{b}_X). \tag{1.5}$$

La sortie est normalisée par une fonction dite «rectified linear unit» [57], pour conserver la signification de détection d'attributs visuels, les nombres négatifs étant difficiles à interpréter dans ce contexte.

Contrairement à [145], aucune hypothèse n'est introduite sur l'espace optimal à atteindre. Nous laissons le problème de transformation de l'espace d'attribut au processus d'optimisation, le critère joint proposé

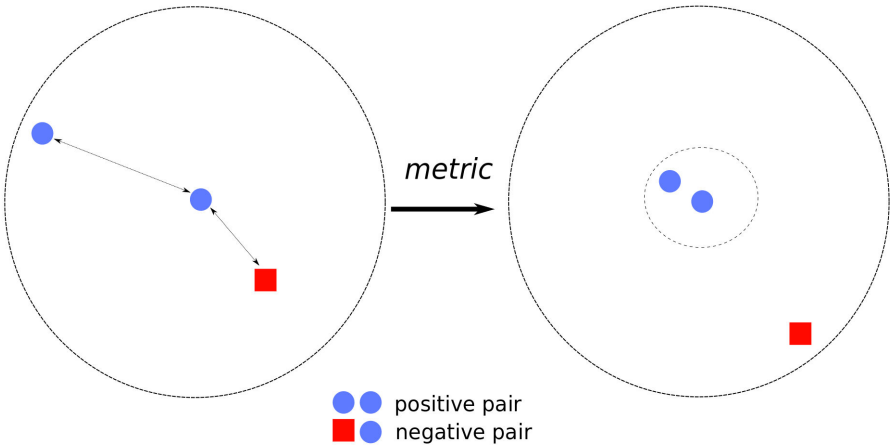


FIGURE 1.10 – Effet de la métrique appliquée à un espace de représentation non optimal. La métrique diminue la distance entre les paires positives (ronds bleus) et maximise la distance entre les paires négatives (carré rouge).

dans la section suivante devrait permettre de trouver cet espace latent optimal.

1.3.4 Apprentissage de métrique

Le problème d'optimisation est maintenant réduit à l'estimation de trois objets mathématiques : le plongement linéaire $\mathbf{W}_X$ de l'image dans l'espace des attributs de dimensions $d \times p$, le biais $\mathbf{b}_X$ de dimension p, et le plongement linéaire de Mahalanobis $\mathbf{W}_A$ de dimensions $p \times m$, m étant un paramètre libre à choisir.

L'approche proposée consiste à construire empiriquement ces objets à partir d'un ensemble d'exemples en appliquant des techniques d'apprentissage statistique. L'ensemble d'apprentissage contient des paires de données $(\mathbf{x}_i, \mathbf{y}_i)$: $\mathbf{x}_i$ est un vecteur de caractéristiques visuelles extraites d'un réseau convolutif profond, alors que $\mathbf{y}_i$ est une description vectorielle basée sur un ensemble d'attributs.

L'objectif de l'apprentissage de métrique est de transformer l'espace

de représentation original de sorte que la métrique résultante prenne en compte la structure statistique des données en utilisant les contraintes de paires. Une façon habituelle de faire est d'exprimer le problème sous forme de classification binaire de paires, où le rôle de la métrique est de rapprocher/séparer les échantillons similaires/dissimilaires (voir figure 1.10).

Il est facile de construire des paires d'exemples similaires et dissimilaires à partir des exemples annotés. Les deux modalités $\mathscr{X}$ et $\mathscr{Y}$ sont échantillonnées aléatoirement et se voient assigner un indicateur $z \in \{-1, 1\}$ indiquant si $\mathbf{y}_i$ est une bonne description d'attribut de $\mathbf{x}_i$ ($z_i = 1$) ou non ($z_i = -1$). Les approches d'apprentissage de métrique tentent de trouver une manière optimale de coder la similarité, en fonction des données.

Nous avons maintenant un jeu de données de triplets $\{(\mathbf{x}_i, \mathbf{y}_i, z_i)\}_{i=1}^{N}$, la variable z indique si les deux modalités sont similaires, c'est-à-dire cohérentes, ou pas. L'étape suivante consiste à décrire un critère empirique qui permettra d'apprendre $\mathbf{W}_X$, $\mathbf{b}_X$ et $\mathbf{W}_A$. L'idée est de décomposer le problème en trois objectifs : l'apprentissage de la métrique, le plongement sémantique et un terme de régularisation.

La partie apprentissage de métrique suit une fonction de coût appelée «hinge loss» désormais standard [139], elle prend la forme suivante pour chaque échantillon :

$$l_H(\mathbf{x}_i, \mathbf{y}_i, z_i, \tau) = \max\left(0, 1 - z_i(\tau - S(\mathbf{x}_i, \mathbf{y}_i)^2)\right). \tag{1.6}$$

Le paramètre supplémentaire τ est un hyperparamètre, dont le rôle est de définir le seuil séparant les exemples similaires et dissimilaires et qui dépend de la distribution des données.

Le critère de plongement dans l'espace des attributs est un critère moindre carré, appliqué uniquement aux données similaires :

$$l_A(\mathbf{x}_i, \mathbf{y}_i, z) = \max(0, z_i).\|\mathbf{y}_i - A_X(\mathbf{x}_i)\|_2^2. \tag{1.7}$$

Son rôle est de s'assurer que la prédiction des attributs est de bonne qualité, de sorte que la différence $\mathbf{y} - A_X(\mathbf{x})$ reflète la dissimilarité due aux incohérences de modalité plutôt qu'à un mauvais plongement sémantique.

La taille du problème d'apprentissage ($d \times p + p + p \times m$) peut être grande et nécessite une régularisation pour éviter un sur apprentissage des données. Nous utilisons une pénalisation quadratique :

$$R(\mathbf{W}_A, \mathbf{W}_X, \mathbf{b}_X) = \|\mathbf{W}_X\|_F^2 + \|\mathbf{b}_X\|_2^2 + \|\mathbf{W}_A\|_F^2 \tag{1.8}$$

où $\|.\|_F$ est la norme de Frobenius.

Le critère d'optimisation global peut maintenant être écrit comme la somme des termes précédemment définis :

$$\begin{aligned}\mathscr{L}(\mathbf{W}_A, \mathbf{W}_X, \mathbf{b}_X, \tau) = \sum_i l_H(\mathbf{x_i}, \mathbf{y_i}, z_i, \tau) + \lambda \sum_i l_A(\mathbf{x_i}, \mathbf{y_i}, z_i) \\ + \mu R(\mathbf{W}_A, \mathbf{W}_X, \mathbf{b}_X)\end{aligned} \tag{1.9}$$

où λ et μ sont des hyperparamètres choisis par validation croisée. À noter que le critère 1.9 peut être interprété comme une approche d'apprentissage multi-objectifs car il mélange deux problèmes (dépendants) : le plongement de l'image dans l'espace d'attributs et la métrique sur cet espace.

Pour résoudre notre problème d'optimisation, nous utilisons une descente de gradient stochastique (voir la section 1.4 pour plus de détails).

1.3.5 Méthodes intelligentes de sélection des paires négatives

Dans un problème d'apprentissage de métrique, alors que l'ensemble des paires positives $\mathbf{D}_+$ est fixé et donné par l'ensemble d'apprentissage, l'ensemble des paires négatives $\mathbf{D}_-$ peut être choisi plus librement ; en effet il y a beaucoup plus de manières d'être différents que similaires, le nombre de paires négatives et positives n'est pas identique. De plus, nous verrons qu'augmenter la taille de $\mathbf{D}_-$ par rapport à $\mathbf{D}_+$ par un facteur n conduit à de meilleurs résultats globaux.

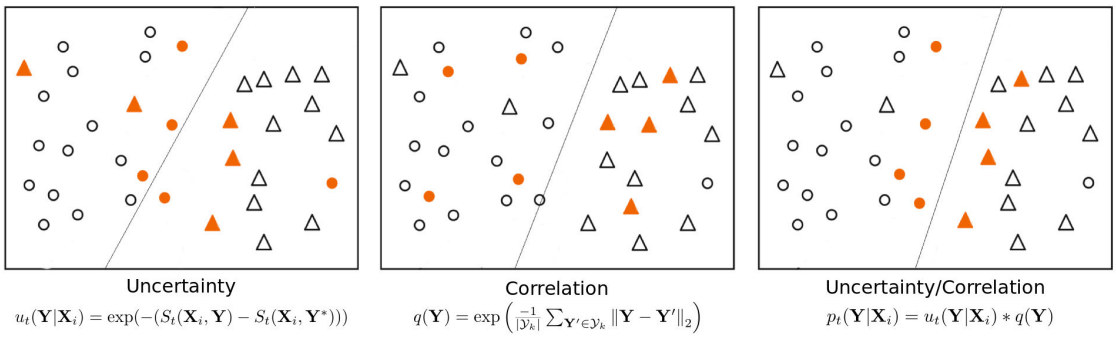


FIGURE 1.11 – Différentes stratégies de sélection des paires négatives. Les symboles orange représentent celles potentiellement sélectionnées par la stratégie. *Incertitude* : sélectionne les exemples proches ou du mauvais côté de la frontière de décision. *Corrélation* : les exemples représentant au mieux une classe sont sélectionnés. *Incertitude / Corrélation* : combine les deux approches, les exemples proches de la frontière de décision ont plus de chance d'être sélectionnés.

Dans ce qui suit, nous explorons trois stratégies pour échantillonner la distribution des paires négatives : nous présentons d'abord une variante de la méthode présentée dans la section précédente puis décrivons deux nouvelles techniques itératives.

1.3.5.1 Sélection aléatoire

Les paires négatives du critère joint (1.9) sont obtenues en associant une image d'apprentissage avec un vecteur d'attribut choisi aléatoirement parmi les autres classes vues. Nous proposons de générer aléatoirement n paires négatives (au lieu d'une) pour une paire positive. Nous incluons dans la fonction objective une pondération pour compenser le déséquilibre entre paires positives et négatives (voir section 1.3.6).

1.3.5.2 Sélection basée sur l'incertitude

Inspirée des techniques utilisées pour la détection d'objets [28, 53, 89, 142], cette stratégie consiste à sélectionner les paires négatives les plus

informatives (en termes d'erreur de compatibilité) (voir figure 1.11).

On note $S_t(\mathbf{x},\mathbf{y})$ le score au temps t. Chaque pas de temps t correspond à une passe d'apprentissage sur l'ensemble du jeu de données. À la première passe, S_1 les paires négatives sont choisies aléatoirement (voir section 1.3.5.1). Puis pour chaque image $\mathbf{x}_i$, les annotations $\mathbf{y}$, provenant des différentes classes vues, sont classées en fonction du score d'incertitude :

$$u_t(\mathbf{y}|\mathbf{x}_i) = \exp(-(S_t(\mathbf{x}_i,\mathbf{y}) - S_t(\mathbf{x}_i,\mathbf{y}^*))) \tag{1.10}$$

où $\mathbf{y}^*$ est le vecteur d'attributs compatible de $\mathbf{x}_i$.

L'idée est que la paire négative la plus similaire à la paire compatible (en termes de score de compatibilité), est la plus pertinente pour améliorer le modèle. C'est la représentation pour laquelle le modèle a le plus de difficulté à prendre sa décision.

Une fois le score d'incertitude associé à toutes les paires négatives, on échantillonne suivant la probabilité de générer la paire en fonction de ce score d'incertitude, cela nous permet de créer l'ensemble $\mathbf{D}_-$. Plus une paire est similaire à la paire compatible (donc avec un score d'incertitude élevé), plus elle a de chance d'être sélectionnée. Ce processus est répété à chaque pas de temps t

1.3.5.3 Incertitude et corrélation pour la sélection des paires négatives

Nous proposons d'améliorer l'approche précédente en prenant en compte la corrélation intra-classe. Le principe sous-jacent qui régit la sélection est que les vecteurs d'attributs les plus corrélés, dans une classe donnée, sont les plus utiles à considérer (voir figure 1.11). La corrélation peut être mesurée par :

$$q(\mathbf{y}) = \exp\left(\frac{-1}{|\mathscr{Y}_k|}\sum_{\mathbf{y}'\in\mathscr{Y}_k}\|\mathbf{y}-\mathbf{y}'\|_2\right) \tag{1.11}$$

où k est l'indice de la classe y et $\mathcal{Y}_k$, l'ensemble des représentations d'attributs.

Un compromis entre l'incertitude et la corrélation est obtenu en utilisant la fonction suivante :

$$p_t(\mathbf{y}|\mathbf{x}_i) = u_t(\mathbf{y}|\mathbf{x}_i) * q(\mathbf{y}) \tag{1.12}$$

où chaque vecteur d'attributs $\mathbf{y}$ à la passe t a un score p_t associé à une image $\mathbf{x}_i$.

De façon similaire à la section 1.3.5.2, l'ensemble des paires négatives de la passe t sont annotées puis échantillonnées selon le score de l'équation 1.12.

1.3.6 Prise en compte du déséquilibre entre paires positives et négatives

Le critère d'apprentissage original 1.9, suppose que les paires négatives et positives sont réparties uniformément. Ce n'est pas le cas dans l'approche proposée : comme préconisé par [62], le critère doit être adapté pour compenser le déséquilibre entre paires positives et négatives.

Nous proposons de pondérer les paires positives et négatives en fonction de leurs fréquences :

$$\begin{aligned}\mathcal{L}(\mathbf{W}_A, \mathbf{W}_X, \mathbf{b}_X, \tau) = &\frac{1}{|\mathbf{D}_+|}\left(\sum_{i\in\mathbf{D}_+} l_H(\mathbf{x_i}, \mathbf{y_i}, z_i, \tau) + \lambda l_A(\mathbf{x_i}, \mathbf{y_i}, z_i)\right) \\ &+ \frac{1}{|\mathbf{D}_-|}\left(\sum_{j\in\mathbf{D}_-} l_H(\mathbf{x_j}, \mathbf{y_j}, z_j, \tau)\right) + \mu R(\mathbf{W}_A, \mathbf{W}_X, \mathbf{b}_X)\end{aligned} \tag{1.13}$$

où $\mathbf{D}_+$ et $\mathbf{D}_-$ correspondent au nombre d'éléments compris dans les ensembles de paires positives et négatives.

1.3.7 Reconnaissance et recherche d'image

Le score de cohérence (1.4) est un outil polyvalent qui peut être utilisé pour plusieurs problèmes d'interprétation d'images. La section 1.4 évalue le potentiel de notre approche sur trois d'entre eux.

1.3.7.1 Classification visuelle sans exemple de référence

L'objectif est de trouver la paire de modalités la plus cohérente étant donné l'image à classer, et un ensemble de descripteurs de classe $\{\mathbf{y}_k^*\}_{k=1}^C$, où k est l'indice d'une classe :

$$k^* = \underset{k \in \{1 \dots C\}}{\arg\min} S(\mathbf{x}, \mathbf{y}_k^*) \tag{1.14}$$

où $\mathbf{y}_k^*$ est une description sémantique par un ensemble d'attributs de la classe d'indice k.

Dans cette formulation, le classement entre les classes C est équivalent à l'identification de la meilleure description d'attributs.

1.3.7.2 Reconnaissance d'image avec un faible nombre d'exemples de référence

La métrique proposée peut facilement être affinée quand de nouvelles données sont disponibles. Une fois le modèle appris avec les données des classes vues, celui-ci est affiné avec de nouveaux triplets $(\mathbf{x}, \mathbf{y}, z)$ provenant des classes non vues. Cela rend possible une approche dite «few-shot», le cadre de décision est identique à celui de l'approche zero-shot.

1.3.7.3 Recherche d'image

L'objectif d'une tâche CBIR est la recherche d'images numériques dans de grandes bases de données, en fonction du contenu de celles-ci.

Le score 1.4 peut être utilisé pour récupérer les données d'une base donnée avec une requête définie par la modalité $\mathbf{y}$:

$$\text{Recherche}(\mathbf{y}, \lambda) = \{\mathbf{x} \in \mathscr{X} / S(\mathbf{x}, \mathbf{y}) < \lambda\} \tag{1.15}$$

La performance est généralement caractérisée par des courbes précision/rappel.

TABLE 1.1 – Statistiques sur les jeux de données.

			# de classes			Nombre d'images				
						Train			Test	
Données	Type	Attr	Y	Y^s	Y^u	Total	Y^s	Y^u	Y^s	Y^u
CUB [154]	fin	312	200	150	50	11 788	7 057	0	1 764	2 967
P&Y [50]	grossier	64	32	20	12	15 339	5 932	0	1 483	7 924
SUN [124]	fin	102	717	645	72	14 340	10 320	0	2 580	1 140
AWA1 [84]	grossier	85	50	40	10	30 475	19 832	0	4 958	5 685
AWA2 [163]	grossier	85	50	40	10	37 322	23 527	0	5 882	7 913

1.4 Expériences

Cette section valide expérimentalement la méthode proposée. Nous introduisons tout d'abord les 5 jeux de données utilisés ainsi que les détails des paramètres expérimentaux. La méthode est évaluée empiriquement sur les trois tâches décrites à la section 1.3.7. Les expériences de classification zero-shot visent à évaluer la capacité du modèle à reconnaître des objets provenant des classes non vues. Cette section évalue également la contribution des différents composants du modèle et compare les résultats aux approches de la littérature. Pour notre deuxième tâche, nous montrons comment notre approche peut servir à l'apprentissage d'un classifieur lorsque seulement quelques échantillons des classes non vues sont disponibles. Finalement, nous évaluons notre modèle sur une tâche de recherche d'images, illustrant la capacité de l'algorithme à retrouver des images à partir d'une requête exprimée par un ensemble d'attributs.

1.4.1 Jeux de données

L'évaluation expérimentale est réalisée sur 5 jeux de données (voir tableaux 1.1) : Caltech-UCSD Birds-200-2011 (CUB) [154], Apascal & Ayahoo (P&Y) [50], SUN Attribute Dataset (SUN) [124], Animals with Attributes (AWA1) [84] et Animals with Attributes 2 (AWA2) [163].

Part	Attributes	Part	Attributes	Part	Attributes
Beak	*HasBillShape, HasBillColor , HasBillLength*	**Back**	*HasBackColor, HasBackPattern*	**Breast**	*HasBreastPattern, HasBreastColor*
Belly	*HasBellyPattern, HasBellyColor*	**Fore-head**	*HasForehead Color*	**Bird** (all parts)	*HasSize, HasShape*
Throat	*HasThroatColor*	**Nape**	*HasNapeColor*	**Head**	*HasHeadPattern*
Crown	*HasCrownColor*	**Eye**	*HasEyeColor*	**Leg**	*HasLegColor*
Tail	*HasUpperTailColor, HasUnderTailColor, HasTailPattern, HasTailShape*	**Wing**	*HasWingPattern, Has WingColor, HasWingShape*	**Body**	*HasUnderpartsColor, HasUpperPartsColor, HasPrimaryColor*

FIGURE 1.12 – Exemples d'images et d'attributs utilisés pour l'évaluation. Chaque jeu de données possède son propre ensemble d'attributs.

Ces jeux de données présentent une grande variété de concepts (voir figure 1.12) : SUN et CUB nécessitent une classification fine et incluent respectivement des images d'oiseaux et de scènes ; AWA1 et AWA2 contiennent des images d'animaux de 50 catégories différentes ; enfin, P&Y représente des concepts plus divers, de la voiture aux animaux.

Pour chaque ensemble de données, des descriptions d'attributs sont fournies, soit au niveau de la classe, soit au niveau de l'image. Les représentations sémantiques sont fournies par image, sous la forme d'attributs binaires pour les données provenant des ensembles P&Y, CUB et SUN. Pour AWA1 et AWA2 la totalité des images d'une classe est représentée par un unique vecteur d'attributs à valeurs continues.

Afin de comparer notre approche avec d'autres travaux de la littérature, nous suivons les mêmes sous-ensembles de test que [163].

1.4.2 Implémentation

Pour permettre une reproduction des travaux présentés, nous consacrons une section dédiée aux détails d'implémentation.

Les caractéristiques image sont extraites à l'aide d'un réseau convolutif ResNet [65] à 101 couches. Nous utilisons l'avant-dernière couche de dimension 2048 comme représentation. Les poids du réseau sont pré-appris sur l'ensemble de données ImageNet [39]. Nous n'appliquons aucun ajustement sur les poids du réseau (sans «fine-tuning»).

Notre modèle possède trois hyperparamètres : λ qui mesure l'importance du plongement sémantique, la dimension de l'espace dans lequel la distance est calculée (m) et le paramètre de régularisation μ. Les valeurs des paramètres sont recherchées grâce à une procédure de validation croisée «zero-shot». 20% des classes vues sont considérées comme non vues (donc utilisées comme ensemble de validation) et permettent de sélectionner les hyperparamètres maximisant la performance sur cet ensemble de données. Une fois les hyperparamètres choisis, l'ensemble d'apprentissage complet est utilisé pour apprendre le modèle final et est évalué sur l'ensemble de test. Nous avons choisi les intervalles de recherche suivants : $\lambda \in [0.05, 1.0]$, $m \in [20\%, 120\%]$ de la dimension initiale des attributs et $\mu \in [0.01, 10.0]$. τ est un paramètre appris pendant l'entraînement.

Les poids des modèles sont initialisés grâce à distribution gaussienne centrée ($\sigma = 0.02$). Ils sont optimisés à l'aide d'une descente de gradient stochastique avec le solveur Adam [78]. Le pas d'apprentissage est de 10^{-4} et la taille des batchs est de 128.

L'une des principales caractéristiques de l'approche par métrique est qu'elle nécessite un ensemble de paires d'images/attributs. Les paires positives et négatives sont obtenues suivant les stratégies présentées dans la section 1.3.5.

TABLE 1.2 – Précision multi-classe en classification zero-shot pour notre modèle MLZSL. Cont. : plongement de l'image dans l'espace initial d'attribut, Métr. : transformation appliquée à l'espace initial d'attribut, Sélect. : stratégie de sélection de paires négatives suivant l'incertitude et la corrélation.

Méthode	CUB	P&Y	SUN	AWA1	AWA2	Avg.
Cont.	51.0	36.3	54.8	55.3	57.5	51.0
Métr.	52.8	38.2	56.1	57.5	59.4	52.7
Métr. + Cont.	53.4	39.0	57.8	57.9	60.5	53.7
Métr. + Cont. + Sélect.	**55.3**	**39.5**	**58.6**	**60.3**	**62.4**	**55.2**

1.4.3 Classification zero-shot

Dans cette première section, nous suivons le protocole standard en classification zero-shot : pendant l'entraînement, seules les données des classes vues sont disponibles pour l'apprentissage des paramètres du modèle. Tandis qu'au moment du test, les nouvelles images (provenant des classes non vues uniquement) doivent être affectées à l'une des classes non vues.

Étude des différents composants Nous commençons tout d'abord pas étudier les différents composants de notre modèle. Comme expliqué dans la section précédente, notre modèle MLZSL est optimisé suivant une fonction de coût multi-critères.

Dans le tableau 1.2 :

- «Cont.» représente notre approche sans métrique, où $W_A = I$. Le modèle est alors un simple plongement linéaire dans l'espace des attributs initiaux.
- «Métr.» est lorsque le terme de prédiction d'attributs est absent du critère, la détection d'attribut n'est pas contrôlée.
- «Métr. + Cont.» représente notre approche avec métrique et contrainte sur la détection d'attributs dans l'image. La sélection des paires

négatives est aléatoire.

- «Métr. + Cont. + Sélect.» avec la méthode de sélection des paires négatives, prenant en compte l'incertitude et la corrélation.

La métrique permet une amélioration notable de 4.2% en moyenne, ceci confirme un de nos objectifs, définir un espace de représentation plus performant.

Il est intéressant d'observer également la baisse de performance lorsque l'un des deux termes est manquant. La métrique étant dépendante de la bonne détection des attributs, elle peut uniquement transformer l'espace initial d'attributs si celui-ci est prédictible.

À la dernière ligne du tableau, nous constatons que la sélection des paires négatives a un impact positif (53.7% vs 55.2%) sur la performance de reconnaissance visuelle.

Dimension de la métrique La figure 1.13 représente le score de classification en fonction de la dimension de la métrique. Le plongement projette les données d'origine dans un espace dans lequel la distance euclidienne est bonne pour la tâche considérée. Cela peut être vu comme un moyen d'exploiter et de sélectionner la structure de corrélation entre les attributs. Nous constatons que les meilleures performances sont généralement obtenues lorsque la dimension de cet espace est inférieure à 40% au nombre d'attributs initial. Cette réduction de l'espace réalise à la fois la sélection et la fusion des attributs entre eux. Cette observation confirme notre première hypothèse, l'ensemble d'attributs défini par l'utilisateur n'est pas optimal pour la tâche de classification zero-shot.

Effet de la sélection des paires négatives sur la métrique La table 1.3 reporte les performances de classification sur l'ensemble de données CUB pour les trois stratégies de sélection, en fonction du ratio d'exemples négatifs pour une paire positive. La configuration «aléatoire» correspond à l'approche initiale. Notre méthode de sélection de paires né-

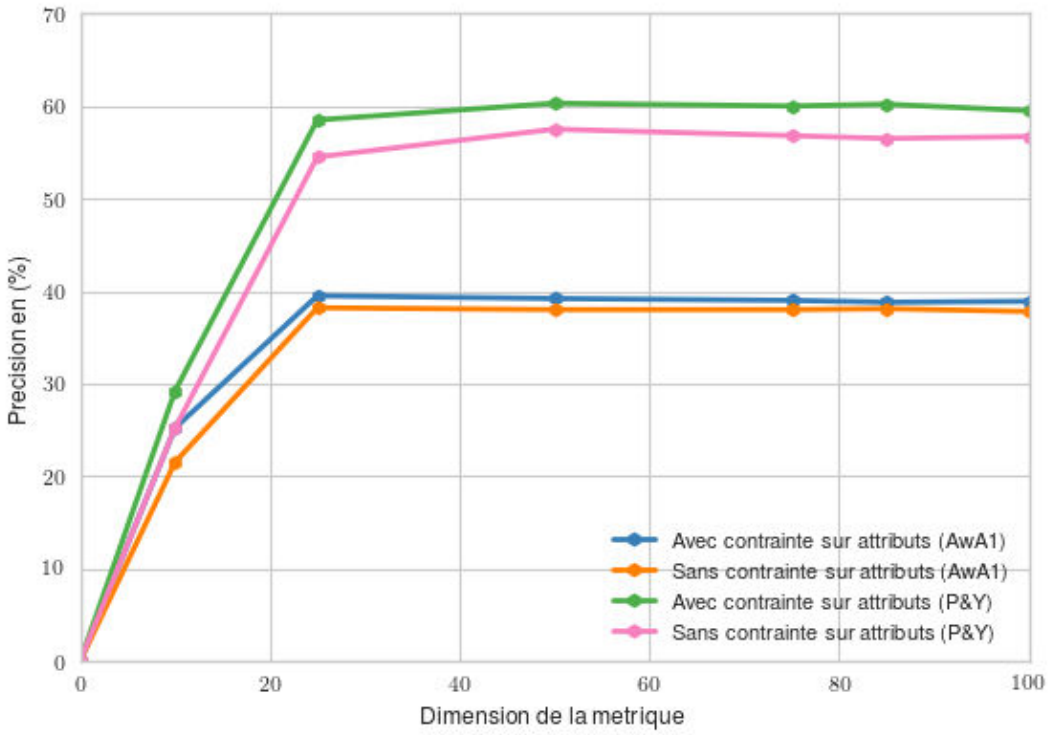


FIGURE 1.13 – Précision zero-shot en fonction de la dimension de la métrique. Les meilleurs résultats sont obtenus quand la dimension est réduite de 40% par rapport à l'espace d'attributs initial. Les scores avec et sans contrainte de plongement sont reportés.

TABLE 1.3 – Précision zero-shot sur le jeu de données CUB en fonction du ratio paire négative/positive et la méthode de sélection de paires utilisée.

Méthode / #nég. paire	1	10	50	100
Aléatoire	53.4	53.5	53.7	53.7
Incertitude	53.9	54.3	54.5	54.6
Incertitude/Corrélation	54.8	55.1	55.3	55.3

gatives permet une amélioration de 1.6%. Prendre en compte la corrélation dans le choix des paires négatives a un impact positif sur la performance de classification (+0.7%). Une des raisons principales est la réduction de la sélection des exemples aberrants (erreur provenant de l'annotation ou exemple non significatif à la classe). On peut aussi noter qu'augmenter le rapport des paires négatives sur les paires positives a une influence

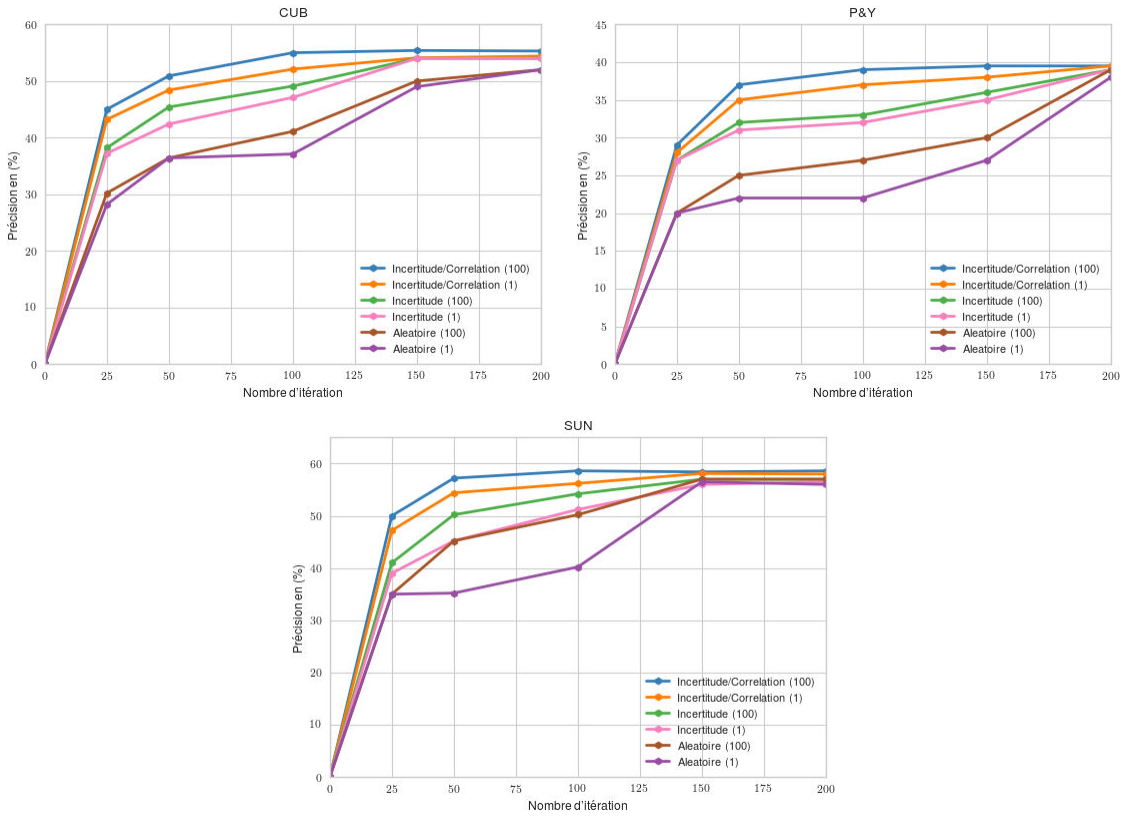


FIGURE 1.14 – Évolution de la performance sur les 3 jeux de données CUB, P&Y et SUN en fonction du nombre d'itérations d'entraînement pour 1 et 100 de ratio. La sélection intelligente de paires négatives permet d'accélérer la convergence du modèle.

favorable sur la précision.

Nous avons également évalué l'impact de la sélection sur la convergence du modèle. La figure 1.14 montre que l'approche Incertitude/Corrélation converge environ 4 fois plus vite que les méthodes basées sur l'Incertitude ou l'aléatoire. Ceci confirme que les paires plus informatives (négatives) sont sélectionnées avec cette stratégie. À noter que le ratio négatif/positif a également un (faible) impact positif sur la convergence.

Comparaison avec l'état de l'art Le tableau 1.4 répertorie les performances, exprimées en tant que précision multi-classes, sur 5 ensembles de données et compare avec les approches de la littérature. Les performances

TABLE 1.4 – Précision multi-classes en classification zero-shot. Les performances de la première partie de tableau proviennent de l'article [163].

	Classification zero-shot					
Méthode	CUB	P&Y	SUN	AWA1	AWA2	Avg.
IAP [84] *PAMI'14*	24.0	36.6	19.4	35.9	35.9	30.4
CMT [145] *NIPS'13*	34.6	28.0	39.9	39.5	37.9	36.0
SAE [80] *CVPR'17*	33.3	8.3	40.3	53.0	54.1	37.8
CONSE [117] *ICLR'13*	34.3	26.9	38.8	45.6	44.5	38.0
DAP [84] *PAMI'14*	40.0	33.8	39.9	44.1	46.1	40.8
SYNC [29] *CVPR'16*	**55.6**	23.9	56.3	54.0	46.6	47.3
SSE [173] *ICCV'15*	43.9	34.0	51.5	60.1	61.0	50.1
LATEM [162] *CVPR'16*	49.3	35.2	55.3	55.1	55.8	50.1
DEVISE [51] *NIPS'13*	52.0	**39.8**	56.5	54.2	59.7	52.4
ESZSL [132] *ICML'15*	53.9	38.3	54.5	58.2	58.6	52.7
SJE [7] *CVPR'15*	53.9	32.9	53.7	**65.6**	61.9	53.6
ALE [6] *PAMI'15*	54.9	39.7	58.1	59.9	**62.5**	55.0
MLZSL	55.3	39.5	**58.6**	60.3	62.4	**55.2**

de la première partie de tableau proviennent de l'article [163]. Quatre types de méthodes y sont répertoriées :

- «classifieurs intermédiaires d'attributs» : DAP [84] et IAP [84] utilisent la représentation sémantique comme couche intermédiaire de la classification ;
- «fonction de compatibilité linéaire» : ALE [6], DEVISE [51], SAE [80], ESZSL [132] et SJE [7] s'expriment sous la forme d'une fonction bilinéaire, qui associe l'information visuelle à celle sémantique ;
- «fonction compatibilité non linéaire» : LATEM [162] et CMT [145] étendent l'approche linéaire en ajoutant un composant de non-linéarité ;
- «modèle hybride» : SSE [173], CONSE [117] et SYNC [29] définissent les classes non vues comme un mélange de classes vues ;

Nous constatons que le top 5 des meilleurs modèles est composé exclusivement d'approches linéaires. Un effet que nous avons remarqué

empiriquement lors de la cross-validation de la métrique. Le faible nombre d'exemples d'apprentissage peut expliquer cette limite de complexité. Sur les cinq jeux de données, notre modèle (ligne MLZSL) atteint des performances moyennes supérieures à celles de l'état de l'art, même comparé à des méthodes plus récentes ([29, 80]).

Les faibles performances de classification sur l'ensemble de données P&Y peuvent s'expliquer par le nombre restreint d'attributs (64) utilisés pour décrire de nombreux concepts très différents (véhicules, animaux, plantes, objets...). De plus les données ont été récoltées par deux groupes de recherche différents, les données de [48] qui sont utilisées pendant l'apprentissage et les données de test qui ont été annotées a posteriori spécifiquement pour la tâche de classification zero-shot [50]. Cette construction en deux étapes est susceptible de biaiser la performance de classification.

1.4.4 Classification avec un faible nombre d'exemples de référence

L'apprentissage avec un faible nombre d'exemples de référence correspond à la situation où un (ou plusieurs) exemple (s) annoté (s) de classes non vues sont disponibles au moment du test. Nous proposons deux approches : *pendant-entraînement (pe) :* les données supplémentaires sont incorporées à l'ensemble d'entraînement et le modèle est optimisé de la même façon que pour une approche zero-shot. *après-entraînement (ae) :* le modèle est d'abord optimisé en utilisant seulement les classes vues (comme avec une approche zero-shot). Une fois cette étape terminée nous introduisons les exemples de données de classe non vues un par un. L'objectif est d'affiner les paramètres du modèle en faisant quelques itérations d'apprentissage sur ces nouvelles données avec un pas d'apprentissage faible (10^{-5}).

La figure 1.15 représente l'évolution de la précision, elle est donnée en fonction du nombre d'images supplémentaires des classes non vues. À

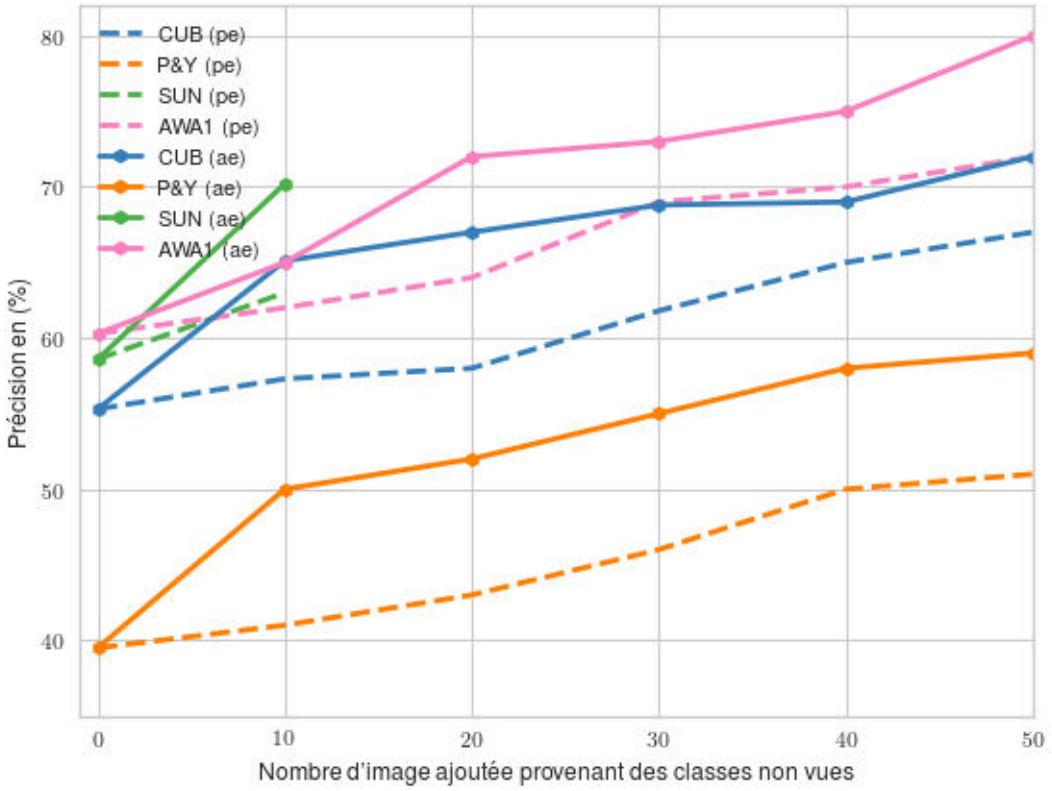


FIGURE 1.15 – Performance de classification avec un faible nombre d'exemples de référence. Précision moyenne (%) en fonction du nombre d'exemples d'apprentissage provenant des classes non vues.

noter que pour l'ensemble de données SUN, nous avons utilisé un maximum de 10 exemples supplémentaires, les classes non vues contiennent seulement 20 images. Nous observons que connaître même un très petit nombre d'exemples annotés améliore significativement la performance.

L'approche *après-entraînement* est la plus performante, elle permet de spécialiser le modèle aux classes non vues uniquement. Contrairement à l'approche *pendant-entraînement*, où on peut observer que les données supplémentaires ont moins d'effet sur la performance finale, car «noyées» avec celles d'entraînements

AWA1 et P&Y sont les ensembles de données profitant au mieux de cet ajout d'information avec une amélioration d'environ 20%/10% pour 50 images ajoutées. C'est un comportement très encourageant pour les

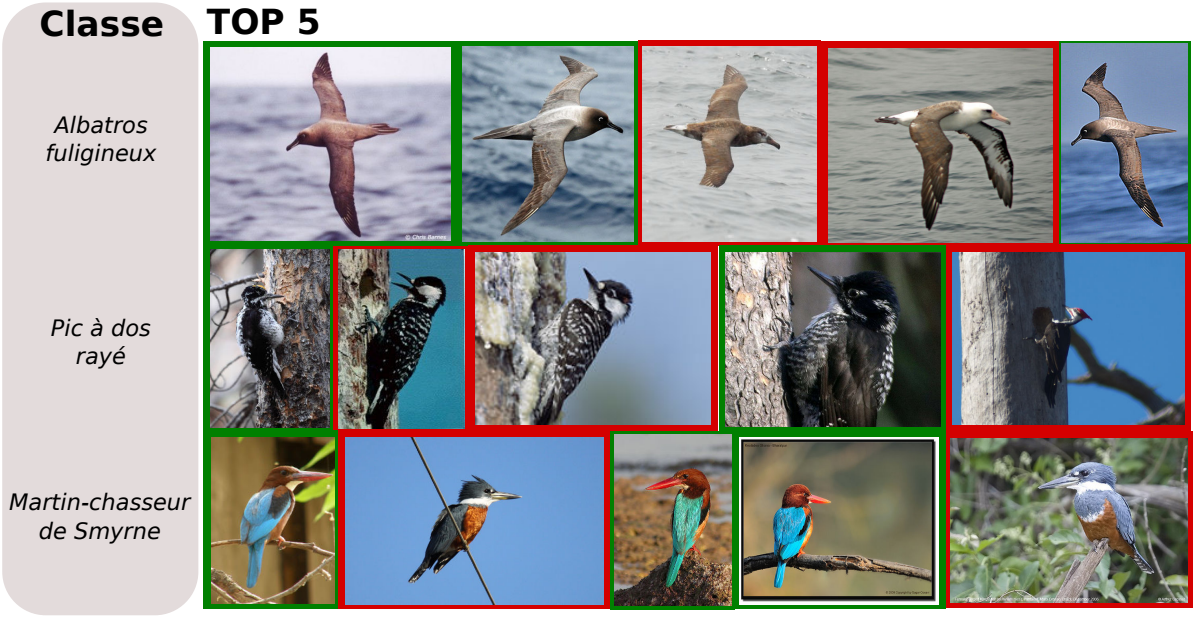


FIGURE 1.16 – Résultats qualitatifs sur l'ensemble de données CUB. Les classes les plus incertaines y sont reportées.

applications à grande échelle où les annotations pour un grand nombre de catégories sont difficiles et coûteuses à obtenir.

1.4.5 Recherche d'images zero-shot

TABLE 1.5 – Recherche d'image zero-shot : précision moyenne (%) sur 4 jeux de données. Les caractéristiques visuelles sont extraites d'un réseau VGG19 [143] à la couche fc7. Pour une comparaison objective, nous suivons les mêmes sous-ensembles que Zhang [173].

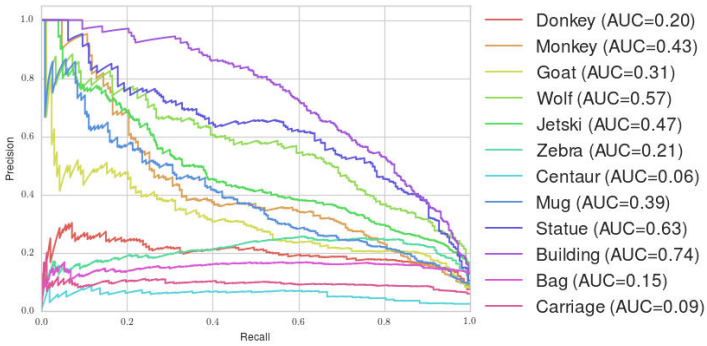
	P&Y	AWA1	CUB	SUN	Av.
SSE [173] *ICCV 2015*	15.4	46.3	4.7	58.9	31.3
JLSE [175] *CVPR 2016*	**38.3**	67.66	**29.15**	**80.01**	**53.78**
MLZSL	36.9	**68.1**	25.3	52.7	45.8

La tâche de recherche d'images zero-shot consiste à chercher dans une base de données des images exposant de nouvelles composantes visuelles (ex : objet) à partir de requêtes exprimées par des attributs. Pour ce faire, notre modèle MLZSL est optimisé de la même façon que pour une

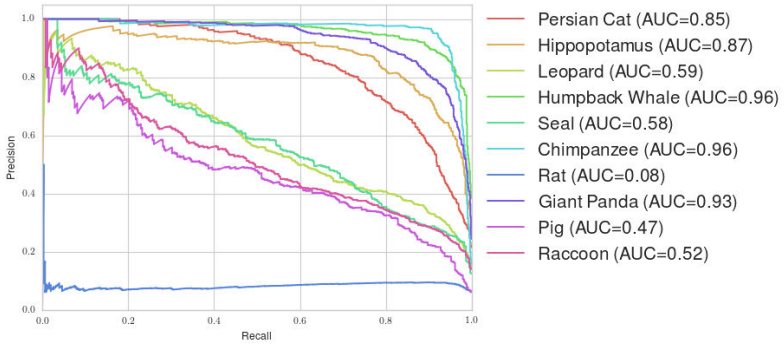
tâche zero-shot standard. Les descriptions d'attributs des classes non vues sont utilisées comme requêtes. Elles permettent de classer les images des classes non vues en fonction de la similarité avec la requête (voir section 1.3.7.3). La table 1.5 reporte la précision moyenne sur 4 ensembles de données. Les caractéristiques visuelles sont extraites d'un réseau VGG19 [143] au niveau de la couche fc7. Pour une comparaison objective, nous suivons les mêmes sous-ensembles que Zhang [173]. Notre modèle surpasse la méthode [173] de plus de 10% en moyenne. Les meilleures performances sont obtenues par [175], cette supériorité peut s'expliquer par le fait que le modèle JLSE est spécifiquement adapté à la tâche de recherche d'images, contrairement à notre approche.

La figure 1.17 montre la précision et le rappel moyen pour chaque classe des 4 ensembles de données. Dans l'ensemble de données P&Y, les classes «donkey», «centaur» et «zebra» ont une très faible précision moyenne. Cela peut s'expliquer par la forte similarité visuelle entre ces classes qui ne diffèrent que de quelques attributs.

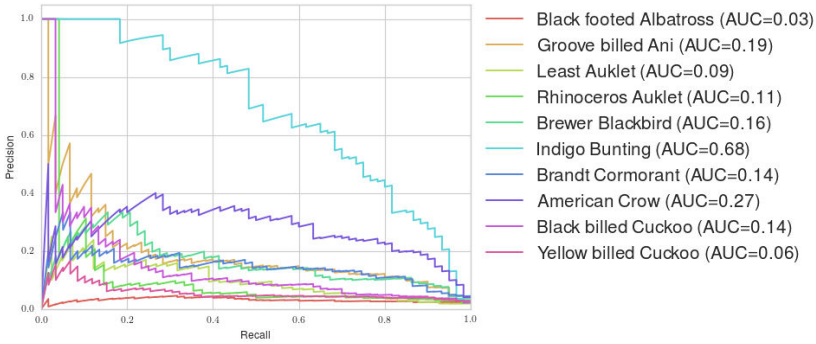
La figure 1.16 reporte un résultat qualitatif sur l'ensemble de données CUB. Des exemples des classes les plus incertaines y sont reportés. Comme pour P&Y, l'ambiguïté provient de la faible différence entre les représentations par attributs.



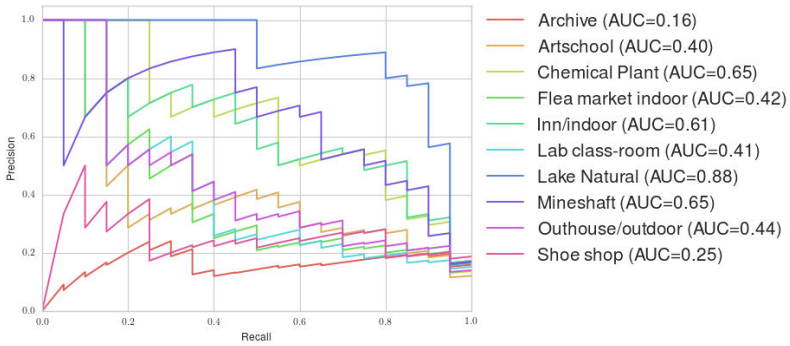
P&Y



AWA1



CUB



SUN

FIGURE 1.17 – Courbes de précision rappel.

1.5 Conclusion

L'objectif de ce chapitre a été la conception et l'évaluation d'une démarche permettant d'introduire de nouveaux modèles d'objets à partir d'une description sémantique dans un système de reconnaissance visuelle. Les approches de classification zero-shot permettent de répondre à cette limitation en autorisant la classification de classes d'objets jamais rencontrées pendant l'apprentissage.

Les classes ne sont plus définies par un ensemble de données de référence, mais par une représentation intermédiaire d'attributs visuels. Ces attributs sont chargés d'exprimer, dans un vocabulaire plus ou moins riche, soit des composants des objets ou actions, soit des configurations globales, soit des caractéristiques intensives ou extensives.

Ce chapitre présente une nouvelle approche pour la classification zero-shot exploitant des techniques d'apprentissage de métriques. Notre approche est basée sur un critère multi-objectifs permettant le plongement de l'image dans l'espace sémantique et la transformation de l'espace initial d'attribut en un nouvel espace optimal pour une tâche de classification zero-shot. Nous proposons également une nouvelle méthode pour la sélection de paires négatives, prenant en compte l'incertitude de prédiction et la corrélation intra-classe. Cette sélection intelligente permet d'améliorer les performances de reconnaissance et la vitesse de convergence du modèle.

Le résultat obtenu peut être utilisé avec polyvalence sur diverses tâches d'interprétation d'images, et montre des performances proches ou supérieures à celles de l'état de l'art sur cinq jeux de données.

En termes de perspectives, les attributs nécessitent un fort effort de supervision, nous verrons dans le chapitre suivant une nouvelle représentation de classe construite sans supervision. Nous nous intéresserons également à l'utilisation de l'espace des caractéristiques visuelles comme espace de décision.

CHAPITRE 2

GÉNÉRATION DE CARACTÉRISTIQUES VISUELLES CONDITIONNÉE PAR REPRÉSENTATION SÉMANTIQUE

Dans le chapitre précédent, nous avons montré qu'un meilleur contrôle de la structure de l'espace d'attribut est nécessaire pour la classification zero-shot (ZSC). Le critère joint basé sur une meilleure détection d'attributs combiné au contrôle de l'espace sémantique a été validé sur 3 tâches zero-shot et a permis d'obtenir des performances à l'état de l'art.

Les méthodes précédemment citées reposent sur l'apprentissage d'un espace d'intégration commun permettant de comparer des caractéristiques visuelles de catégories inconnues avec des descriptions sémantiques. Ces approches sont limitées, car i) l'utilisation de fonctions de compatibilité souffre du «hubness problem» [141]. ii) Les tâches de classification avec des images provenant de catégories vues et non vues pendant la phase de test (zero-shot généralisée ou GZSC) ne peuvent pas être traitées efficacement [30].

Ce chapitre suggère d'aborder les tâches de ZSC et GZSC en :

- apprenant un générateur conditionnel de caractéristiques visuelles ;
- générant des exemples d'entraînement artificiels pour les catégories sans exemples ;
- optimisant un classifieur discriminant grâce aux exemples d'entraînement vrais et artificiels.

Le problème de ZSC est alors transformé en un problème d'apprentissage supervisé standard.

Des expériences avec 4 modèles génératifs et 6 ensembles de données valident expérimentalement l'approche, donnant des résultats à l'état de l'art en ZSC et GZSC.

2.1 Introduction

Les algorithmes de zero-shot récents suivent un mécanisme de décision commun. Pour une image $\mathbf{x}$, la classe y choisie est celle maximisant le score de compatibilité de la fonction : $f(\mathbf{x}) = \arg\max_y S(\mathbf{x}, y)$. La fonction de compatibilité, pour sa part, est souvent définie comme $S(\mathbf{x}, y; \mathbf{W}) = \theta(\mathbf{x})^t \mathbf{W} \phi(y)$ où θ et ϕ sont deux projections et $\mathbf{W}$ une fonction bilinéaire reliant les deux modalités dans un plongement commun.

Il existe différentes variantes dans la littérature récente sur la façon dont les plongements ou la mesure de similarité sont calculés [29, 39, 51, 132, 158, 162, 173], mais dans tous les cas, la classe est choisie comme celle maximisant le score de compatibilité.

Cette approche d'intégration et de compatibilité maximale n'exploite cependant pas, dans la phase d'apprentissage, les informations potentiellement contenues dans la représentation sémantique des catégories non vues. La seule étape où une capacité de discrimination est exploitée est la sélection d'étiquettes finales qui utilise un schéma de décision $\arg\max_y$, la fonction de compatibilité étant optimisée avec des données provenant

des classes vues uniquement.

Ce type d'approche souffre de plusieurs défauts :

Hubness problem. Les techniques zero-shot présentées précédemment sont sujettes au phénomène appelé le «hubness problem» [141]. Ce phénomène est un problème intrinsèque des espaces de grande dimension : à mesure que la dimensionnalité de l'espace augmente, un certain nombre d'éléments (dans notre cas, un certain nombre de classes d'objets), qui ne sont pas du tout similaires à tous les autres, deviennent des «hubs», ils peuvent apparaître comme proches voisins de nombreuses images. À l'inverse on nomme les «anti-hubs» des points éloignés de toute donnée. L'émergence de ces «hubs» diminue l'utilité de la recherche par similarité, car la liste retournée contiendra souvent les mêmes classes, quelle que soit l'image de requête. Les «hubs> nuisent alors à la précision des prédictions.

Shigeto [141] montre que le «hubness problem» ne se produit pas seulement à cause de la dimensionnalité de l'espace, mais également en raison du type de projection utilisée par les approches zero-shot. La norme des objets projetés (images) a tendance à être inférieure à celle des objets cibles (classes), phénomène accentué avec l'ajout d'un terme de régularisation. C'est pourquoi il propose d'effectuer le plongement inverse qui est traditionnellement réalisé, projetant les labels dans l'espace visuel. L'hypothèse étant que l'espace visuel suit une meilleure distribution pour la prise de décision d'une classification.

Les travaux présentés dans ce chapitre vont dans ce sens, nous proposons de prendre la décision de classification dans l'espace des caractéristiques visuelles. De plus, notre approche n'est pas sujette au phénomène de «hubs», car la décision est faite grâce à un classifieur discriminant.

Biais dans la décision de classification. En classification zero-shot on différencie deux types de classes : vues et non vues, où les exemples étiquetés ne sont disponibles que pour les classes vues. Les techniques

présentées précédemment sont évaluées en fonction de leur capacité à distinguer les classes non vues, en supposant l'absence d'objets vus pendant la phase de test.

Cette hypothèse ne reflète pas la réalité, les classes vues étant souvent plus communes que les classes non vues. Il est donc irréaliste de supposer que nous ne les rencontrerons jamais pendant la phase de test. Pour que les modèles zero-shot soient vraiment utiles, ils doivent discriminer avec précision des images provenant de classes vues ou non (GZSC).

Chao [30] montre empiriquement que les approches de la littérature ne permettent pas d'obtenir des performances de classification acceptables. En particulier, les images de test des classes non vues sont presque toujours classées en tant que classes vues. Ce comportement est dû à l'existence d'un biais dans la fonction de décision, la principale cause étant que la fonction de compatibilité est optimisée avec des images de classes vues uniquement.

L'approche présentée dans ce chapitre permet en partie de répondre à cette problématique. Après génération de caractéristiques visuelles pour toutes les classes non vues, celles-ci sont utilisées, en combinaison avec les images d'entraînement des classes vues, pour l'apprentissage du classifieur discriminant. L'information des deux types de classes ayant été vue pendant l'apprentissage, le biais de décision est alors réduit.

Dans ce chapitre nous proposons de générer des caractéristiques visuelles à partir d'une représentation sémantique. Un générateur conditionnel est entraîné à générer des exemples d'entraînement artificiels pour les catégories non vues. Une fois générées ces caractéristiques artificielles peuvent être utilisées avec des techniques de classification discriminante classiques de la littérature (ex : SVM).

Nous validons notre approche sur 6 ensembles de données, notamment un test à grande échelle sur ImageNet.

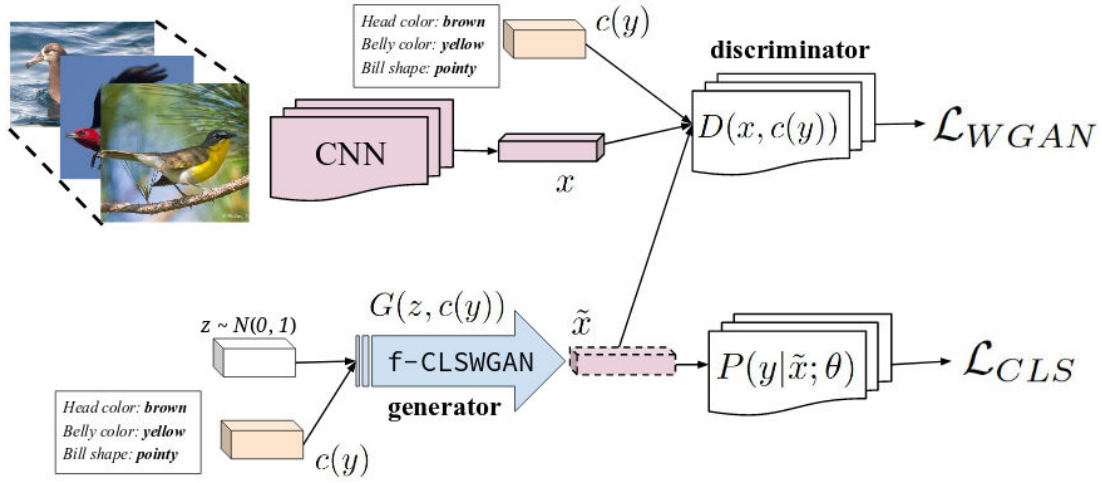


FIGURE 2.1 – *Figure extraite de [163] :* génération de caractéristiques visuelles pour les classes non vues grâce à un «Generative Adversarial Network», conditionné par une représentation d'attributs. Un critère de classification sur les caractéristiques générées permet d'affiner le résultat.

2.2 État de l'art

2.2.1 Génération d'exemples de référence

Comme remarqué par [62], l'apprentissage à partir d'un ensemble de données déséquilibrées, où le nombre d'exemples d'une classe (majoritaire) est beaucoup plus élevé que les autres, représente un défi important pour la communauté de l'apprentissage automatique. Lorsque les ensembles d'apprentissage sont déséquilibrés, plusieurs travaux de la littérature abordent la question de générer de nouveaux exemples d'apprentissage à partir de ceux existants.

Après estimation de la moyenne et la variance de l'ensemble d'entraînement, [106] génère des exemples artificiels à partir d'une approximation gaussienne de la distribution des données d'apprentissage. Pour contrer l'effet négatif d'un ensemble de données déséquilibrées, [15] adapte la technique SMOTE («Synthetic Minority Over-sampling Technique»), et

génère de nouvelles instances à partir d'un exemple choisi et de son voisin le plus proche. La génération de nouveaux exemples d'entraînement à partir des exemples existants est également au cœur de la technique appelée «Data Augmentation», fréquemment utilisée pour l'optimisation des réseaux de neurones profonds appliqués à des images[87]. Les approches préconisées sont les transformations horizontales, verticales et même la rotation.

Les techniques mentionnées ne peuvent pas être utilisées lorsque pour certaines catégories aucune donnée d'entraînement n'est disponible. Cependant, dans ces situations, si une représentation paramétrique sous-jacente des données peut être obtenue (ex. : attributs), elle peut être utilisée pour générer de nouvelles données d'apprentissage, en supposant un lien entre la représentation sous-jacente et l'espace visuel. [46] explore cette stratégie en générant de manière synthétique des données pour la tâche de détection de logo de marque. Des images générées de manière synthétique sont obtenues en appliquant des transformations géométriques et photométriques à des exemples. Une idée similaire est également présentée dans [73] pour la détection de texte dans des images. Dans [26], les auteurs capturent d'abord ce qu'ils appellent «The Gist of a Gesture» en enregistrant des gestes humains, puis en les représentant par un modèle cinématique et en utilisant ce modèle pour générer un grand nombre de gestes réalistes.

Récemment, [58] propose une nouvelle approche appelée «Generative Adversarial Network» (GAN), un type de modèle génératif qui capture une distribution de données, telles que des images, à partir d'un domaine particulier. Les GAN consistent à estimer une distribution via un processus adversaire, en entraînant simultanément deux modèles. Un modèle génératif qui capture la distribution des données, et un modèle discriminant qui estime la probabilité qu'un échantillon provienne des données d'apprentissage plutôt que du générateur. L'optimisation du réseau se fait sous forme d'un jeu «min max», où le générateur doit tromper le discriminant,

TABLE 2.1 – L'apprentissage zero-shot se présente suivant différentes configurations. X : image Y_s : ensemble des classes vues, Y_u : ensemble des classes non vues, $\{X_{i:n}\}_u$: ensemble d'images non étiquetées.

	Entraînement	Test	Objectif
Vanilla	$y_s \in Y_s$	$y_u \in Y_u$	$f : X \rightarrow Y_u$
Généralisé	$y_s \in Y_s$	$y \in Y_s \cup Y_u$	$f : X \rightarrow Y_s \cup Y_u$
Transductif	$y \in Y_s + \{X_{i:n}\}_u$	$y_u \in Y_u$	$f : X \rightarrow Y_u$

entraîné à détecter les vrais exemples des faux. Le GAN a également été étendu au conditionnement, soit avec une étiquette de classe [111], soit par des descriptions textuelles [130], introduit à la fois dans le générateur et le discriminant.

D'autres types de modèles génératifs existent. [18] propose de générer des images grâce à une architecture auto-encodeur, en altérant l'entrée pour améliorer l'invariance au bruit. [102] étend cette approche avec une version plus performante grâce à un mécanisme adversaire, dans le but de contraindre le code. Finalement [91] propose le «Generative Moment Matching Network», un modèle avec une architecture simple, «feedforward» et une fonction de coût ayant pour but de minimiser l'écart entre les moments de deux distributions.

Récemment [11, 164] proposent une solution proche de la méthode présentée dans ce chapitre, ils utilisent un modèle génératif pour générer des caractéristiques visuelles pour les classes non vues (voir figure 2.1). Contrairement à nos travaux, aucune étude quantitative sur la nature du générateur et son architecture n'est réalisée.

2.2.2 De l'approche transductive à généralisée

Toutes les approches mentionnées précédemment considèrent que la fonction de plongement et de compatibilité doit être apprise à partir d'un ensemble de données d'apprentissage de classes connues et utilisées pour déduire l'étiquette des images de classes non vues.

Cependant, un problème différent peut être résolu lorsque les images des classes inconnues sont déjà disponibles au moment de l'apprentissage et peuvent donc être utilisées pour produire un meilleur plongement (voir tableau 2.1). Dans ce cas, le problème peut être interprété comme un problème d'apprentissage transductif. Les données des classes non vues sont utilisées sous la forme d'une supervision partielle.

Dans [100], les images non étiquetées sont utilisées pour améliorer la détection d'attributs et ainsi obtenir une meilleure généralisation aux exemples des classes non vues. Alors que [145] considère l'introduction de classes non vues comme un problème de détection de nouveauté dans un cadre de classification multi-classes. Si l'image est d'une catégorie connue, un classifieur standard peut être utilisé. Sinon, les images sont attribuées à une classe non vue en fonction de son score de compatibilité. En utilisant l'analyse de corrélation canonique multi-vues, Fu [52] introduit un processus d'adaptation de domaine entre les données des classes vues et non vues, en les projetant dans un espace latent. Li [88] apprend de façon semi-supervisée un plongement, en considérant conjointement la classification multi-classes des classes vues et non-vues. Dans le but de surmonter le problème de changement de domaine, [79] propose un codage épars, régularisé par un ensemble de données non annotées du domaine cible.

La tâche de zero-shot généralisée, proposée par [30], suppose que les catégories vues et non vues sont présentes au moment du test. Le problème de perte de performance, dont souffrent les approches traditionnelles, est dû à la fonction de compatibilité favorisant les images provenant des classes vues, car optimisée sur ces étiquettes. L'approche par génération, présentée dans ce chapitre, permet de limiter cet effet, car elle utilise des exemples d'entraînement (artificiels) de classes vues et non vues pendant l'optimisation, évitant les problèmes susmentionnés.

The bird has a white underbelly, black feathers in the wings, a large wingspan, and a white beak.

This bird has distinctive-looking brown and white stripes all over its body, and its brown tail sticks up.

This flower has a central white blossom surrounded by large pointed red petals which are veined and leaflike.

Light purple petals with orange and black middle green leaves

FIGURE 2.2 – *Figure extraite de [7] :* exemples de descriptions textuelles de l'ensemble de données CUB [154] et FLOWER [116].

2.2.3 Informations sémantiques hétérogènes

Les approches citées précédemment se concentrent sur une représentation sémantique d'attributs. Comme mentionné dans le chapitre précédent, trouver un ensemble discriminant et significatif d'attributs peut parfois être difficile, la représentation peut être jugée trop rigide et non naturelle.

D'autres choix possibles sont proposés dans la littérature. Les paramètres $\mathbf{W}$ de la fonction de compatibilité peuvent être prédits à partir d'une description textuelle sémantique de l'image. À titre d'exemple, [47] a pour but de prédire les paramètres d'un classifieur pour une nouvelle catégorie basée uniquement sur une description textuelle de cette catégorie. La description des catégories non vues se présente sous la forme d'un texte tel qu'un article d'encyclopédie, sans qu'il soit nécessaire de définir explicitement les attributs. Le problème est résolu grâce à l'introduction d'une fonction de régression, apprise du domaine de caractéristiques textuelles au domaine de classification visuelle. [14] s'appuie sur cette idée et utilise une fonction de régression basée sur un réseau neuronal profond. Le modèle proposé fournit un moyen de générer automatiquement une liste de pseudo-attributs pour chaque catégorie visuelle constituée de mots

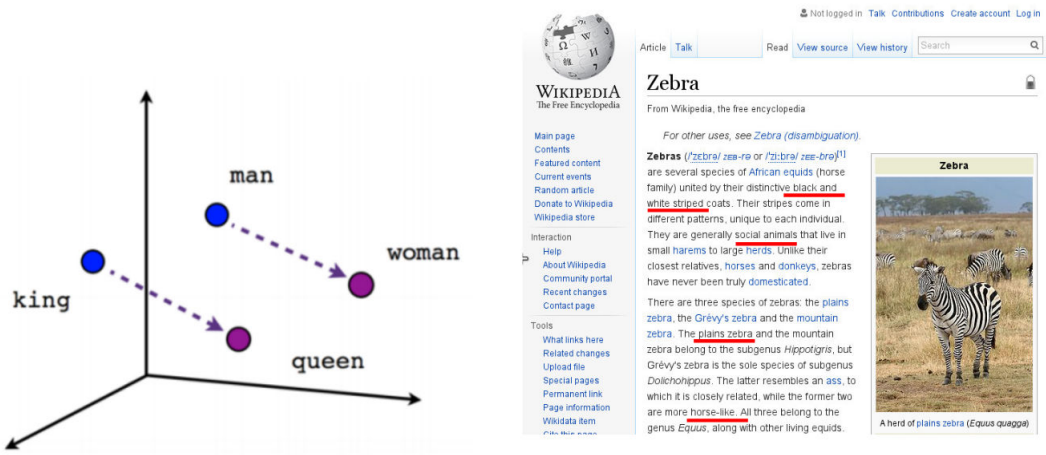


FIGURE 2.3 – Gauche (*figure extraite de [110]*) : exemple de résultat obtenu avec une approche «word2vec», la sémantique y est respectée. Droite : article Wikipédia utilisé pour représenter sémantiquement la classe zèbre.

provenant d'articles de Wikipédia.

Même si un article Wikipédia contient peu d'information visuelle, les méthodes basées sur ce type de représentation fonctionnent étonnamment bien pour la reconnaissance visuelle zero-shot, mais il existe toujours un écart important entre les méthodes par description et les attributs annotés par l'humain. Pour limiter cet écart, [7] propose de décrire chaque image par une description textuelle, beaucoup plus naturelle dans sa construction (voir figure 2.2). Cette approche est actuellement la plus performante bien qu'elle nécessite un effort de supervision important pour la création de l'ensemble d'entraînement.

L'espace des noms de classe peut également constituer un ancrage intéressant, comme dans les travaux de [54, 107, 117, 173] qui représente des images en tant que mélanges de distributions de classes connues. Dans [173] les classes non vues sont définies en tant que proportions des classes vues, la décision de classification est réalisée suivant la similarité des classes observées.

Récemment une approche a gagné en popularité, notamment sur des

problèmes à grande échelle, comme pour l'ensemble de données ImageNet [39]. L'utilisation de technique de type «word2vec» (Skipgram [110], Glove [125]) permet d'extraire pour chaque classe une représentation sémantique basée sur son nom. La figure 2.3 montre que la sémantique est conservée : deux mots sémantiquement proches sont représentés par des vecteurs proches par l'encodage. [29, 51] sont les premiers a montrer que l'approche est possible, malgré des performances plus faibles que pour une représentation supervisée.

FIGURE 2.4 – Projection t-SNE 2D [99] de la représentation d'attributs (gauche) et des caractéristiques visuelles extraites d'un réseau convolutif ResNet [65] (droite). Cette figure confirme notre intuition, l'espace des caractéristiques visuelles est mieux organisé, plus structuré.

2.3 Génération de caractéristiques visuelles pour l'ensemble des classes non vues

2.3.1 Approche discriminante

En classification zero-shot, aucune donnée n'est disponible pour les classes non vues. L'approche adoptée dans cette partie est de générer artificiellement des données pour les classes non vues à partir de leur représentation sémantique. Grâce à un transfert sémantique, les classes vues et leurs représentations sémantiques permettent l'optimisation d'un générateur d'exemple. Nous générons, non pas les pixels des images, mais les caractéristiques visuelles extraites d'un réseau convolutif profond (ex. dernière couche d'un ResNet [65]). Après génération, un modèle de classification discriminant (ex. : machine à vecteurs de support) permet de prédire le label d'image provenant de classes vues et non vues.

La disponibilité des données pour les classes non vues présente deux

avantages : cela permet la classification d'images provenant des classes vues et non vues comme un processus homogène, un unique modèle, quelle que soit la nature de l'image. La tâche de zero-shot généralisée est alors considérée comme un problème de classification supervisée - ce qui devrait permettre de distinguer plus efficacement les classes ;

Cela permet également d'effectuer une inférence de classification directement dans l'espace des entités visuelles plutôt que dans un espace sémantique. Les données sont naturellement plus facilement séparables dans le premier espace (car optimisées pour), en particulier lorsqu'on utilise des caractéristiques profondes. La figure 2.4 montre une projection t-SNE 2D [99] des deux types de représentations et confirme cette intuition.

Soit $\widehat{\mathscr{D}}_u = \{\hat{\mathbf{x}}_i^u, \mathbf{a}_i^u, y_i^u\}_{i=1}^{N_u}$ un ensemble de données générées à partir des représentations sémantiques des classes non vues $\mathbf{a}^u \in \mathscr{A}_u$, où $\hat{x}_i^u$ est une caractéristique visuelle artificielle d'une des classes non vues y^u. La fonction de classification zero-shot peut s'écrire fonctionnellement : $\hat{y} = f_D(\mathbf{x}; \widehat{\mathscr{D}}_u, \mathscr{D}_s)$ et peut être utilisée en association avec les données vues $\mathscr{D}_s$, pour apprendre avec supervision un modèle de classification.

2.3.2 Génération de caractéristiques visuelles

Les modèles de génération présentés dans cette section s'appuient sur les approches récemment proposées pour la génération de données conditionnelles, présentées dans la section 2.2.1. L'idée est d'apprendre globalement un processus génératif aléatoire paramétrique G en utilisant un critère différentiable capable de comparer une distribution de données cible et une distribution générée.

Étant donné $\mathbf{z} \in \mathbb{R}^d$ un échantillon aléatoire d'une distribution multivariée, typiquement uniforme ou gaussienne, et $\mathbf{W}$ l'ensemble des paramètres : de nouveaux exemples de données cohérents avec la description sémantique $\mathbf{a}$ sont générés en appliquant la fonction : $\hat{\mathbf{x}} = G(\mathbf{a}, \mathbf{z}; \mathbf{W})$. Un

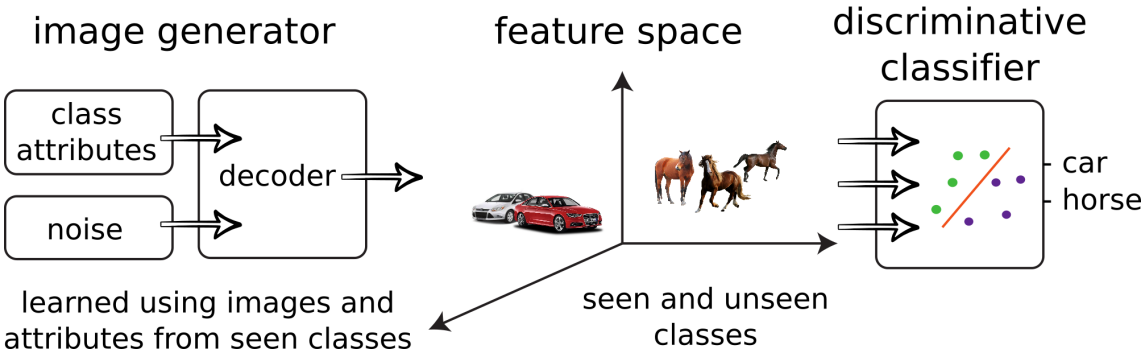


FIGURE 2.5 – La génération de caractéristique image $\hat{\mathbf{x}}$ est conditionnée par : la représentation sémantique $\mathbf{a}$ et un échantillon aléatoire d'une distribution multivariée $\mathbf{z}$. Après génération, un modèle de classification $\hat{y} = f_D(\mathbf{x}; \widehat{\mathscr{D}}_u, \mathscr{D}_s)$ peut être optimisé avec supervision pour les classes vues et non vues.

moyen simple de générer des données conditionnelles $\hat{\mathbf{x}}$ est de concaténer la représentation sémantique $\mathbf{a}$ et l'échantillon aléatoire $\mathbf{z}$ comme entrée d'un réseau, tel qu'indiqué sur la figure 2.5.

Nous présentons maintenant 4 stratégies différentes pour concevoir un tel générateur de données conditionnel, la structure fonctionnelle du générateur étant commune à toutes les approches décrites.

2.3.3 Generative Moment Matching Network

Une première approche consiste à adapter le «Generative Moment Matching Network» (GMMN) proposé dans [91] au conditionnement.

Le GMMN est un réseau de neurones permettant d'échantillonner à partir d'une distribution simple (ex. : gaussienne ou uniforme), des échantillons plus complexes (ex. : la distribution des caractéristiques visuelles). Le réseau de neurones est alors génératif et conditionné par un simple a priori $\mathbf{z} \in \mathbb{R}^d$ à partir duquel il est facile de prélever des échantillons. Celui-ci est propagé de façon déterministe à travers les couches cachées du réseau, la sortie est un échantillon de la distribution cible. Le critère

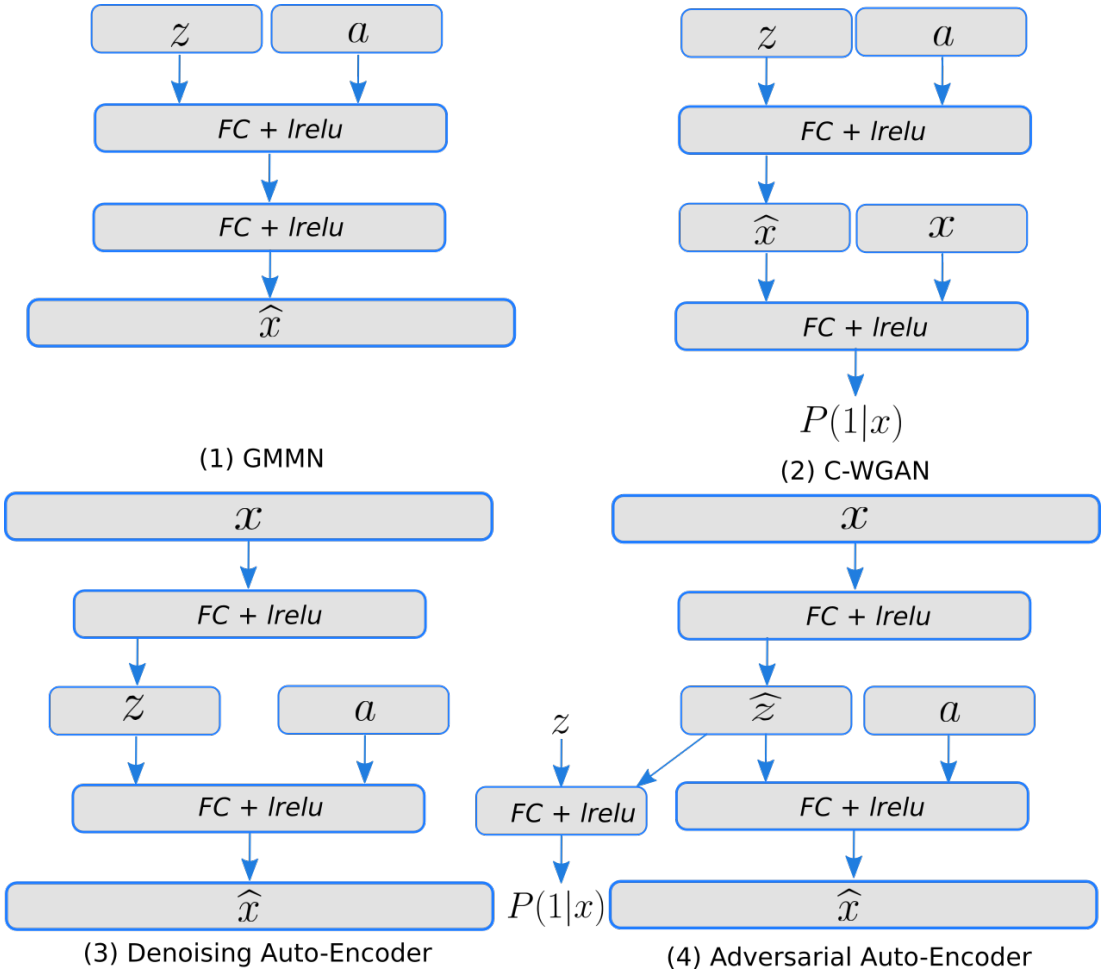


FIGURE 2.6 – Architecture des modèles génératifs proposés. $\mathbf{z}$: bruit gaussien, $\mathbf{a}$: description sémantique, FC + lrelu : couche entièrement connectée suivie d'une fonction non linéaire type «Leaky ReLUs» [98], $\hat{\mathbf{x}}$: caractéristiques visuelles générées.

d'optimisation est inspiré d'une technique de test d'hypothèse statistique connue sous le nom de «Mean Maximum Discrepancy» (MMD) [60]. Les échantillons générés sont contraints de suivre les moments statistiques de l'ensemble des données cibles.

Pour permettre la génération de caractéristiques visuelles, considérant la classe cible, nous proposons une version conditionnée de cette approche. Le réseau génératif est conditionné par deux entrées : $\mathbf{a} \in \mathscr{A}$ est la présentation sémantique continue ou binaire d'une classe d'objet, $\mathbf{z} \in \mathbb{R}^d$

correspond à un échantillon d'une distribution gaussienne multivariée de dimension d. L'utilisation de $\mathbf{z}$ permet de la variabilité dans le processus de génération, pour une même représentation sémantique.

Ces deux entrées sont concaténées pour former un vecteur $\mathbf{h} = [\mathbf{z}; \mathbf{a}]$, utilisé comme entrée du réseau neuronal G :

$$\widehat{\mathbf{x}} = G(\mathbf{h}; \mathbf{W}) = \underset{n}{\text{lrelu}}(...(\underset{1}{\text{lrelu}}(\mathbf{h}^T \mathbf{W_1})...)\mathbf{W_n})\mathbf{W_{n+1}} \tag{2.1}$$

où G est la fonction génératrice, constituée de plusieurs couches entièrement connectées suivies d'une fonction non linéaire de type «Leaky ReLUs» [98]. $\mathbf{W}$ représente les paramètres du réseau. Un exemple d'architecture pour G est illustré sur la figure 2.6 (1), le réseau possède deux couches entièrement connectées non linéaires et une couche de sortie de la dimension des caractéristiques cibles.

Le processus génératif sera considéré comme bon si pour toute description sémantique $\mathbf{a}$, la distribution des exemples d'entraînements $\mathscr{D}_s$ et la population $\widehat{\mathscr{X}}$ échantillonnée à partir du générateur ont une faible mesure de divergence de probabilité. Cette divergence peut être approchée en utilisant un noyau de Hilbert [60] - typiquement une combinaison linéaire de fonctions gaussiennes avec différentes variances :

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{-1}{2\sigma^2} \left\| \mathbf{x} - \mathbf{x}' \right\|^2\right) \tag{2.2}$$

où σ est la largeur du noyau.

Le grand avantage de k est d'être différentiable, il peut donc être utilisé comme une fonction d'apprentissage :

$$L_{\text{A-GMMN}}(\mathbf{a}) = \frac{1}{N_a^2} \sum_{i=1}^{N_a} \sum_{i'=1}^{N_a} k(\mathbf{x}_i^a, \mathbf{x}_{i'}^a) + \frac{1}{M_a^2} \sum_{j=1}^{M_a} \sum_{j'=1}^{M_a} k(\widehat{\mathbf{x}}_j^a, \widehat{\mathbf{x}}_{j'}^a) - \frac{2}{N_a M_a} \sum_{i=1}^{N_a} \sum_{j=1}^{M_a} k(\mathbf{x}_i^a, \widehat{\mathbf{x}}_j^a) \tag{2.3}$$

où $\mathbf{x}^a$ (resp. $\widehat{\mathbf{x}}^a$) est un échantillon réel (resp. généré) d'une classe y représentée par la description sémantique $\mathbf{a}$. N_a et M_a correspondent aux nombres d'éléments réels et générés.

Les paramètres du réseau $\mathbf{W}$ sont alors obtenus par descente de gradient stochastique, en utilisant les données générées et réelles conditionnées par la description sémantique $\mathbf{a}$. Après apprentissage, la génération d'un échantillon $\hat{\mathbf{x}}$ se fait à partir de la description sémantique $\mathbf{a}$ et de $\mathbf{z}$ gaussien.

2.3.4 Wasserstein Generative Adversarial Network

Notre second modèle s'appuie sur les principes des réseaux adversaires [58], notamment le «Wasserstein Generative Adversarial Network» (WGAN) [10].

Deux réseaux sont placés en compétition dans un scénario de théorie des jeux. Le premier réseau est le générateur, il génère un échantillon (ex. une image), tandis que son adversaire, le discriminant essaie de détecter si un échantillon est réel ou bien s'il est le résultat du générateur. Pour approximer la distribution p_g des données $\mathbf{X}$, on définit une variable de bruit d'entrée sur $p_z(\mathbf{z})$ car plus facile à échantillonner. Puis on définit notre générateur $G(\mathbf{z};\mathbf{W_g})$, où G est une fonction différentiable représentée par un réseau neuronal multi-couches de paramètres $\mathbf{W_g}$. Nous définissons également un deuxième réseau neuronal multi-couches $D(\mathbf{x};\mathbf{W_d})$ qui produit un seul scalaire. $D(\mathbf{x})$ représente la probabilité que $\mathbf{x}$ provienne des données réelles plutôt que de p_g.

Le discriminant D est optimisé pour maximiser la probabilité d'une vraie image, provenant des données $\mathbf{X}$ et minimiser la probabilité d'une image générée. Au contraire G cherche à tromper D, CAD : que l'image générée se voit assigner une forte probabilité d'être vraie de la part de D.

D et G, maximisent la fonction de coût approximant la distance de Wasserstein :

$$L_S = \mathbb{E}_{x\sim p_{data}(\mathbf{x})}[D(\mathbf{x})] - \mathbb{E}_{z\sim p_z(z)}[D(G(\mathbf{z}))] - \lambda E\left[(\|\nabla_{\hat{x}} D(\hat{\mathbf{x}})\|_2 - 1)^2\right] \tag{2.4}$$

le troisième terme contraint le gradient d'avoir une norme unitaire.

Nous proposons de conditionner la génération du WGAN (voir figure 2.6), noté C-WGAN. Les distributions générées et vraies sont comparées en utilisant la distance de Wasserstein, et la qualité de la génération conditionnelle est contrôlée par un coût de classification entre l'ensemble des classes cibles. L'entrée du réseau génératif $G(\mathbf{h};\mathbf{W_g})$ est conditionnée par $\mathbf{h} = [\mathbf{z};\mathbf{a}]$. $\mathbf{z} \in \mathbb{R}^d$ est un échantillon d'une distribution gaussienne multivariée de dimension d et $\mathbf{a} \in \mathscr{A}$ correspond à la représentation sémantique continue d'une classe d'objet.

Un deuxième processus de conditionnement est introduit dans la fonction de coût, G et D sont optimisés à classifier des exemples dans leur classe d'objets. On a alors :

$$L_C = \mathbb{E}[\log P(C = c \mid \mathbf{x})] + \mathbb{E}[\log P(C = c \mid G(\mathbf{h}))] \tag{2.5}$$

G est entraîné à maximiser $L_S + L_C$, D à minimiser $L_C - L_S$.

Ce modèle présente des similarités avec le modèle GMMN, la principale différence étant que les distributions du GMMN des données vraies et générées sont comparées en utilisant les statistiques empiriques basées sur un ensemble de noyaux alors que dans le cas C-WGAN, elles sont mesurées par un modèle paramétrique discriminant appris sur les données elles-mèmes.

2.3.5 Denoising Auto-Encoder

Notre troisième générateur s'appuie sur le travail présenté dans [18], où une structure encodeur/décodeur est proposée pour concevoir un générateur de données.

Un auto-encodeur se compose toujours de deux parties, l'encodeur et le décodeur. L'encodeur est en charge de transformer la donnée d'entrée $\mathbf{x} \in \mathscr{X}$ en une représentation minimale, le code $\mathbf{z} \in \mathbb{R}^d$. Cette représentation est généralement de plus faible dimension pour permettre l'apprentissage d'un espace de représentation latent optimal en termes d'information.

Le décodeur, à partir du code $\mathbf{z}$ essaye de reconstruire la donnée d'entrée. Cette reconstruction se doit d'être le plus fidèle à l'entrée, le décodeur se comporte comme un générateur de représentation.

Un «denoising auto-encoder» est une version modifiée de l'auto-encodeur classique qui prend une entrée partiellement corrompue et apprend à reconstruire l'entrée originale débruitée. L'ajout de bruit dans le processus d'encodage permet d'obtenir un modèle plus robuste aux données corrompues.

Nous avons développé une extension capable de contrôler la génération de données, en concaténant la représentation sémantique $\mathbf{a}$ au code qui est fourni par l'encodeur (voir figure 2.6). En pratique, ce modèle est appris comme un auto-encodeur standard, sauf que i) du bruit est ajouté à l'entrée et ii) la représentation sémantique $\mathbf{a}$ est concaténée au code dans la couche cachée centrale.

Pendant l'apprentissage le code $\mathbf{z}$ est obtenu grâce à l'encodeur E :

$$\mathbf{z} = E(\tilde{\mathbf{x}}; \mathbf{W}_E) = \underset{n}{\text{lrelu}}(...(\underset{1}{\text{lrelu}}(\tilde{\mathbf{x}}^T \mathbf{W}_{\mathbf{E_1}})...)\mathbf{W}_{\mathbf{E_n}})\mathbf{W}_{\mathbf{E_{n+1}}} \tag{2.6}$$

où $\tilde{\mathbf{x}}$ est une caractéristique visuelle corrompue grâce à un bruit gaussien. E est la fonction encodeur, constituée de plusieurs couches entièrement connectées suivies d'une fonction non linéaire de type «Leaky ReLUs» [98]. $\mathbf{W}_E$ représente les paramètres du réseau.

Le code $\mathbf{z}$ est concaténé à la représentation sémantique $\mathbf{a}$, $\mathbf{h} = [\mathbf{z}; \mathbf{a}]$ et permet la génération de caractéristiques visuelles. Le vecteur $\mathbf{h}$ est ensuite utilisé comme entrée du réseau neuronal D pour conditionner la génération :

$$\hat{\mathbf{x}} = D(\mathbf{h}; \mathbf{W}_D) = \underset{n}{\text{lrelu}}(...(\underset{1}{\text{lrelu}}(\mathbf{h}^T W_{D_1})...)W_{D_n})W_{D_{n+1}} \tag{2.7}$$

Le modèle est optimisé à minimiser l'erreur quadratique de reconstruction :

$$\zeta(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \tag{2.8}$$

Pour générer de nouveaux exemples, la partie décodeur est utilisée. Le code $\mathbf{z}$ est échantillonné à partir d'une distribution gaussienne multivariée et est concaténé à la représentation sémantique $\mathbf{a}$ de la classe souhaitée, cela permet de générer les caractéristiques visuelles $\widehat{\mathbf{x}}$.

2.3.6 Adversarial Auto-Encoder

Notre dernier générateur est inspiré de [102], qui est une extension de l'approche encodeur/décodeur (voir section 2.3.5).

Comme expliqué précédemment, l'encodeur est en charge de transformer la donnée d'entrée bruitée $\tilde{\mathbf{x}}$ en une représentation minimale, le code $\mathbf{z}$. Cependant la distribution du code latent $\mathbf{z}$ produit n'est pas connue. Nous n'avons aucune garantie que cette distribution soit facile à échantillonner, certains sous-espaces de $\mathbf{z}$ peuvent ne correspondre à aucune caractéristique visuelle cible. [102] propose de contraindre $\mathbf{z}$ à suivre une distribution donnée (ex. : gaussienne). Cette contrainte supplémentaire doit garantir que toutes les parties de l'espace du code $\mathbf{z}$ produiront des données significatives.

À la manière d'un «Generative Adversarial Network» (voir section 2.3.4), la contrainte sur le code $\mathbf{z}$ est proposée sous forme adversaire. Un réseau additionnel appelé discriminant est en charge de distinguer entre un échantillon généré de la distribution cible ou le code intermédiaire $\mathbf{z}$ provenant de l'auto-encodeur. Le discriminant agit comme un mécanisme de régularisation forçant $\mathbf{z}$ à suivre la distribution cible. La méthode proposée peut alors être vue comme deux réseaux distincts avec l'auto-encodeur en charge de reconstruire la donnée d'entrée bruitée et la partie GAN contraignant le code.

Le décodeur de l'auto-encodeur permet de générer des caractéristiques visuelles :

$$\widehat{\mathbf{x}} = D(\mathbf{h}; \mathbf{W}_D) = \underset{n}{\text{lrelu}}(...(\underset{1}{\text{lrelu}}(\mathbf{h}^T \mathbf{W}_{\mathbf{D_1}})...)\mathbf{W}_{\mathbf{D_n}})\mathbf{W}_{\mathbf{D_{n+1}}} \tag{2.9}$$

où $\mathbf{h} = [\mathbf{z}; \mathbf{a}]$ est la concaténation du code et la représentation sémantique. $\mathbf{z} = E(\tilde{\mathbf{x}}; \mathbf{W}_E)$ est le code généré par l'encodeur à partir de la donnée d'entrée bruitée.

La partie auto-encodeur est optimisée suivant l'erreur quadratique de reconstruction $\zeta(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2$

Le deuxième réseau correspond à la partie adversaire. Tout comme le «Generative Adversarial Network» celui-ci est composé d'un générateur G et d'un discriminant Ds. Le générateur partage la même architecture et les mêmes poids que l'encodeur $\mathbf{W}_G = \mathbf{W}_E$. L'objectif du générateur est de tromper le discriminant en produisant des échantillons les plus similaires à $\mathbf{Z} \sim \mathcal{N}(\mu, \sigma^2)$:

$$\min_G \max_{Ds} L = \mathbb{E}_{z \sim p_{gauss}(z)}[\log Ds(\mathbf{z})] + \mathbb{E}_{x \sim p_{data}(x)}[\log(1 - Ds(G(\tilde{\mathbf{x}})))] \tag{2.10}$$

La fonction de décision binaire du discriminant, vrai ou faux échantillon, est un classifieur linaire.

Pendant l'apprentissage, l'auto-encodeur et la partie adversaire sont optimisés simultanément. Pour générer de nouveaux exemples, seule la partie décodeur est utilisée.

2.3.7 Classification de caractéristiques générées

Étant donné $\mathbf{a}_i^u$ la représentation d'une classe non vue, on peut maintenant synthétiser de nombreuses caractéristiques visuelles $\mathbf{x}$. Après avoir répété ce processus de génération de caractéristiques pour chaque classe non vue, nous obtenons un ensemble d'apprentissage synthétique $\widehat{\mathscr{D}}_u = \{\hat{\mathbf{x}}_i^u, \mathbf{a}_i^u, y_i^u\}_{i=1}^{N_u}$. Nous apprenons ensuite de façon supervisée un classifieur discriminant «softmax» sur l'ensemble de données $\hat{y} = f_D(\mathbf{x}; \widehat{\mathscr{D}}_u, \mathscr{D}_s)$. Nos caractéristiques générées permettent d'entraîner ces méthodes en association avec les données vues $\mathscr{D}_s$.

Nous utilisons un classifieur «softmax» standard optimisé suivant le log de vraisemblance :

$$\min_{\theta} -\frac{1}{|D|} \sum_{(\mathbf{x_i}, y_i) \in D}^{|D|} \log P(y \mid \mathbf{x}; \theta) \tag{2.11}$$

où $\theta \in \mathbb{R}^{d_x \times |N|}$ est la matrice de poids d'une couche entièrement connectée qui mappe les caractéristiques visuelles $\mathbf{x}$ sur l'espace des N classes. Les scores sont normalisés suivant une fonction «softmax» :

$$P(y \mid \mathbf{x}; \theta) = \frac{\exp(\theta_y^T \mathbf{x})}{\sum_i^N \exp(\theta_i^T \mathbf{x})} \tag{2.12}$$

La fonction de décision est alors : $f(\mathbf{x}) = \arg\max_y P(y \mid \mathbf{x}; \theta)$, y pouvant être une classe vue ou non vue.

TABLE 2.2 – Statistiques sur le jeu de données ImageNet [39]. Y : nombre de classe total, Y^s : nombre de classe vue, Y^s : nombre de classe non vue. *2-hop*, *3-hop* et *All* correspondent aux différents sous-ensembles de test.

	Nombre de classes			Nombre d'images				
				Train			Test	
[39]	Y	Y^s	Y^u	Total	Y^s	Y^u	Y^s	Y^u
2-hop	2 509	1 000	1 509	3 287 021	1 281 167	0	45 000	1 960 854
3-hop	8 678	1 000	7 678	8 224 142	1 281 167	0	45 000	6 897 975
All	21 345	1 000	20 345	14 197 122	1 281 167	0	45 000	12 860 955

2.4 Expériences

Dans un premier temps, cette section présente les ensembles de données et les paramètres expérimentaux. Puis nous évaluons notre approche sur trois tâches zero-shot : classification zero-shot standard, généralisée et à grande échelle. Ceci nous permet de comparer les différents modèles génératifs décrits dans la section précédente, ainsi que l'approche générative par rapport aux travaux des l'état de l'art.

2.4.1 Jeux de données

L'évaluation expérimentale est réalisée sur les 5 jeux de données présentés dans le chapitre précédent 1.4.1 ainsi que sur un nouveau jeu de données de grande taille pour évaluer le passage à l'échelle (voir tableaux 1.1 et 2.2) : Caltech-UCSD Birds-200-2011 (CUB) [154], Apascal & Ayahoo (P&Y) [50], SUN Attribute Dataset (SUN) [124], Animals with Attributes (AWA1) [84], Animals with Attributes 2 (AWA2) [163] et ImageNet [39].

Comme pour le chapitre précédent, nous suivons les mêmes sous-ensembles de test que [163], afin de comparer nos approches avec les autres travaux de la littérature.

ImageNet est un jeu de données proposant un large spectre de classes différentes, organisées suivant une hiérarchie de concepts. Pour ImageNet

[39], l'approche ne diffère, contrairement aux autres ensembles de données, aucune représentation d'attribut n'étant fournie pour définir les classes non vues. Nous représentons ces catégories en utilisant un modèle de langage skip-gram [110]. Ce modèle est appris sur l'ensemble des articles de l'encyclopédie Wikipédia (≈ 3 milliards de mots). Skip-gram est un modèle de langage capable de prédire le contexte (les n mots suivants et précédents) d'un mot. Le réseau de neurones est composé d'une couche d'entrée, une couche cachée et une couche de sortie ayant la taille du vocabulaire (même taille que la couche d'entrée). La couche cachée est composée de 500 neurones dans notre implémentation.

Pour chaque classe, nous utilisons la couche cachée comme représentation sémantique. Certaines classes ne peuvent pas être représentées, car leur label ne figure pas dans le vocabulaire établi par l'analyse du corpus Wikipédia. Ces classes sont ignorées, ce qui porte le nombre de classes de 20 842 à 20 345. Pour une comparaison équitable, nous prenons le même modèle de langage que [29] avec les mêmes classes exclues. Comme dans [29, 51] nos modèles sont évalués sur trois scénarios différents, avec un nombre croissant de classes non vues : i) *2-hop* : 1509 classes ii) *3-hop* : 7.678 classes, iii) *All* : toutes les catégories non vues.

2.4.2 Implémentation

Pour permettre une reproduction des travaux présentés, nous consacrons une section dédiée aux détails d'implémentation.

Les caractéristiques visuelles sont extraites à l'aide d'un réseau ResNet [65] à 101 couches ; nous utilisons les mêmes représentations visuelles que [163]. Nous utilisons la couche supérieure de dimension 2048 comme représentation. Les poids du réseau sont pré appris sur l'ensemble de données ImageNet [39], optimisés pour une tâche de classification d'objets. Nous n'appliquons aucun ajustement sur les poids du réseau.

Chaque architecture possède son propre ensemble d'hyper-paramètres

(typiquement le nombre d'unités par couche, le nombre de couches cachées, le taux d'apprentissage, etc.). Leurs valeurs sont obtenues par «validation croisée zero-shot» (voir procédure décrite dans le chapitre 1). Une fois les hyper-paramètres réglés, l'ensemble d'apprentissage complet est utilisé pour apprendre le modèle final et est évalué sur l'ensemble de test. Les valeurs typiques pour le nombre de neurones (respectivement le nombre de couches cachées) sont comprises entre l'intervalle $[500-2000]$ (respectivement 1 ou 2). Le classifieur après génération de caractéristiques visuelles est une couche totalement connectée avec une fonction de décision softmax. Le coût est un critère entropique. Dans toutes nos expériences, nous générons 500 images artificielles par classe, ce que nous considérons comme un compromis raisonnable entre la performance et le temps d'entraînement ; nous n'observons aucune amélioration significative en ajoutant plus d'images.

Les paramètres des modèles sont initialisés suivant une distribution gaussienne centrée ($\mu = 0.0, \sigma = 0.02$) et optimisés avec une descente de gradient stochastique à l'aide du solveur Adam [78]. Le taux d'apprentissage est de 10^{-4}, en utilisant des batch de taille 128, sauf pour le GMMN où chaque batch contient toutes les images d'entraînement d'une classe, pour rendre l'estimation des statistiques plus fiable. Afin d'éviter le sur-apprentissage, nous utilisons du dropout [146] à chaque couche (probabilité d'activation de 0.8 pour les couches d'entrées et de 0.5 pour les couches cachées). Les données d'entrée (à la fois les caractéristiques visuelles et les vecteurs w2c) sont normalisées entre $[0, 1]$ par transformation affine.

2.4.3 Classification zero-shot

Dans cette première section, nous suivons le protocole standard en classification zero-shot : pendant l'entraînement, seules les données des classes vues sont disponibles pour l'apprentissage des paramètres du

TABLE 2.3 – Précision en classification zero-shot. Les performances de la première partie de tableau proviennent de l'article [163].

	Classification zero-shot					
Méthode	CUB	P&Y	SUN	AWA1	AWA2	Avg.
IAP [84] *PAMI'14*	24.0	36.6	19.4	35.9	35.9	30.4
CMT [145] *NIPS'13*	34.6	28.0	39.9	39.5	37.9	36.0
SAE [80] *CVPR'17*	33.3	8.3	40.3	53.0	54.1	37.8
CONSE [117] *ICLR'13*	34.3	26.9	38.8	45.6	44.5	38.0
DAP [84] *PAMI'14*	40.0	33.8	39.9	44.1	46.1	40.8
SYNC [29] *CVPR'16*	55.6	23.9	56.3	54.0	46.6	47.3
SSE [173] *ICCV'15*	43.9	34.0	51.5	60.1	61.0	50.1
LATEM [162] *CVPR'16*	49.3	35.2	55.3	55.1	55.8	50.1
DEVISE [51] *NIPS'13*	52.0	39.8	56.5	54.2	59.7	52.4
ESZSL [132] *ICML'15*	53.9	38.3	54.5	58.2	58.6	52.7
SJE [7] *CVPR'15*	53.9	32.9	53.7	65.6	61.9	53.6
ALE [6] *PAMI'15*	54.9	39.7	58.1	59.9	62.5	55.0
f-CLSWGAN [164] *CVPR'18*	57.3	-	60.8	68.2	-	-
RG-GZSL [11] *CVPR'18*	**59.6**	-	**63.3**	69.5	69.2	-
MLZSL	55.3	39.5	58.6	60.3	62.4	55.2
AE	53.7	33.9	54.2	69.4	69.4	56.1
Ad. AE	53.4	37.9	58.5	68.2	70.3	57.7
C-WGAN	56.3	40.3	59.9	68.0	**70.4**	59.0
GMMN	59.4	**41.7**	60.1	**70.7**	69.9	**60.4**

modèle. Tandis qu'au moment du test, les nouvelles images (provenant des classes non vues uniquement) doivent être affectées à l'une des classes non vues.

Comparaison avec les approches à l'état de l'art Le tableau 2.3 rassemble les performances de précision multi-classes de notre approche et les compare avec les modèles à l'état de l'art de la littérature. Comme dans le chapitre précédent, nous reprenons les résultats de l'article [163].

Les approches génératives (les modèles proposés dans la section précédente et f-CLSWGAN/RG-GZSL) permettent un gain notable en précision

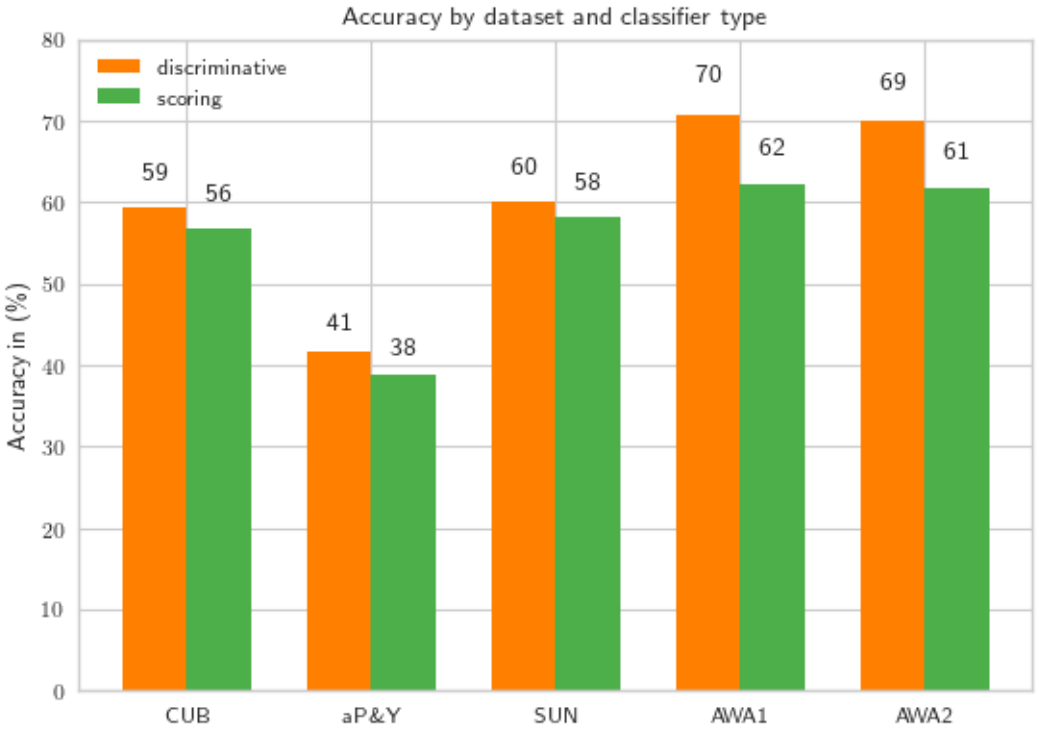


FIGURE 2.7 – GMMN - classifieur discriminant (softmax) vs fonction de similarité (MLZSL) sur la tâche de classification zero-shot.

multi-classes (par rapport aux méthodes de plongement sémantique) sur l'ensemble des jeux de données, avec une amélioration moyenne de plus de 5%.

La figure 2.7 nous permet de comprendre la raison de cette hausse de performance, elle représente la précision de classification du modèle génératif GMMN suivant le type de fonction de décision utilisé après la génération des caractéristiques visuelles (softmax ou métrique). L'approche par métrique est la même que présentée dans le chapitre précédent, cependant l'apprentissage est effectué avec les caractéristiques visuelles générées des classes non vues.

Utilisée avec une fonction de compatibilité, l'approche par génération permet d'obtenir des performances similaires au modèle du chapitre 1 (55.0% vs 55.2%). Alors que couplée avec un modèle discriminant, la précision de classification est en hausse de 4.8% en moyenne. La nature de

FIGURE 2.8 – Visualisation T-SNE des caractéristiques visuelles générées par le GMMN, C-WGAN, AE et ADV-AE. Nous les comparons aux «vraies» caractéristiques visuelles. Les couleurs représentent les différentes classes de l'ensemble de données AWA2. *(à visualiser en version numérique)*

la fonction de décision semble alors être une des raisons de l'augmentation des performances des méthodes par génération.

Comparaison des différents modèles génératifs Nous souhaitons comparer les performances des 4 modèles génératifs décrits dans la section 2.3.2, sur les tâches de classification zero-shot. Les performances sont indiquées dans le tableau 2.3. Nous pouvons remarquer que le modèle

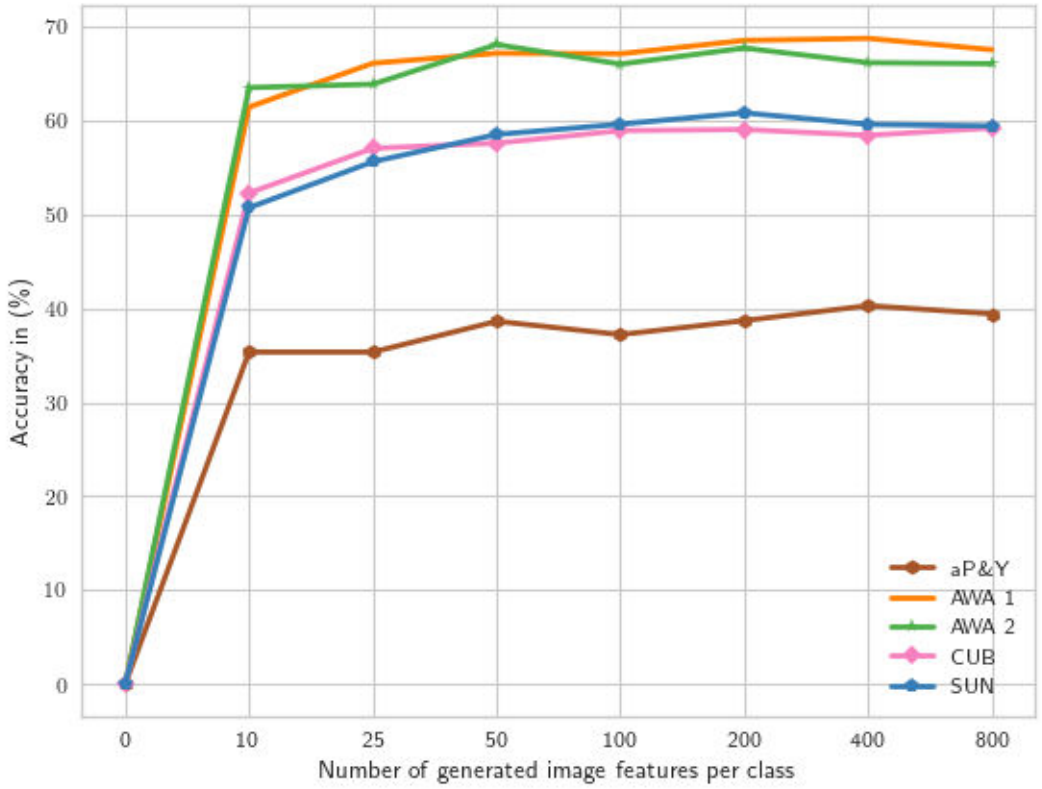


FIGURE 2.9 – GMMN - Évolution de la performance en fonction du nombre de caractéristiques visuelles générées par classe, sur la tâche de classification zero-shot.

GMMN surpasse en moyenne les 3 autres approches. Son optimisation est également plus stable et rapide que les autres versions.

Afin de mieux comprendre le comportement des 4 modèles, nous projetons en 2D les données générées avec une approche T-SNE [99]. Les projections sont données figure 3.1. Pour chaque modèle, 10 caractéristiques visuelles par classes non vues sont générées et comparées à 10 caractéristiques visuelles réelles. Chaque exemple est annoté selon le modèle génératif utilisé et coloré suivant la classe d'objet représentée. Les caractéristiques générées par le modèle GMMN semblent plus rapprochées, donc plus similaires, aux données réelles que les autres approches, suivant le critère de réduction de dimension de l'algorithme T-SNE.

Nous expliquons la supériorité du modèle GMMN par la nature de sa

fonction de coût, basée sur la divergence des distributions alors que les générateurs AE, ADV-AE et C-WGAN doivent l'apprendre. De plus pour le «denoising auto-encodeur» aucune contrainte n'est introduite pour garantir que le code $\mathbf{z}$ soit défini sur l'ensemble des valeurs échantillonnables.

Influence du nombre de caractéristiques générées Pour toutes nos expériences, nous avons généré 500 caractéristiques visuelles par classe. Comme confirmé sur la figure 2.9, l'ajout d'exemples supplémentaires ne permet aucune amélioration de performance significative.

2.4.4 Classification zero-shot généralisée

Dans cette section, nous suivons le protocole zero-shot généralisé («Generalized Zero-Shot Learning») introduit par Chao etal [30]. Comme mentionné précédemment, les données de test proviennent de toutes les classes, vues ou non vues. Cette tâche est plus réaliste et plus difficile, car le nombre de classes candidates est plus grand.

Dans le tableau 2.4 nous suivons les notations de [163] :

— u est la précision de classification des images de test de classes non vues ;
— s est la précision de classification des images de test des classes vues ;
— H indique la moyenne harmonique entre u et s.

Dans les deux cas, le classifieur est appris avec des données d'apprentissage combinant des caractéristiques visuelles vraies pour les classes vues et générées pour les non-vues.

La table 2.4 montre que les approches génératives surpassent de loin toutes les autres approches qui reposent sur un plongement sémantique.

TABLE 2.4 – Classification zero-shot généralisée. u est la précision de classification des images de test de classes non vues, s est la précision de classification des images de test des classes vues et H indique la moyenne harmonique entre u et s.

	Zero-shot généralisé														
	CUB			P&Y			SUN			AWA1			AWA2		
Méthode	u	s	H	u	s	H	u	s	H	u	s	H	u	s	H
IAP [84] *PAMI'14*	0.2	**72.8**	0.4	5.7	65.6	10.4	1.0	37.8	1.8	2.1	78.2	4.1	0.9	87.6	1.8
CMT [145] *NIPS'13*	7.2	49.8	12.6	1.4	85.2	2.8	8.1	21.8	11.8	0.9	87.6	1.8	0.5	90.0	1.0
CMT* [145] *NIPS'13*	4.7	60.1	8.7	10.9	74.2	19.0	8.7	28.0	13.3	8.4	86.9	15.3	8.7	89.0	15.6
SAE [80] *CVPR'17*	7.8	54.0	13.6	0.4	80.9	0.9	8.8	18.0	11.8	1.8	77.1	3.5	1.1	82.2	2.2
CONSE [117] *ICLR'13*	1.6	72.2	3.1	0.0	**91.2**	0.0	6.8	39.9	11.6	0.4	88.6	0.8	0.5	**90.6**	1.0
DAP [84] *PAMI'14*	1.7	67.9	3.3	4.8	78.3	9.0	4.2	25.1	7.2	0.0	**88.7**	0.0	0.0	84.7	0.0
SYNC [29] *CVPR'16*	11.5	70.9	19.8	7.4	66.3	13.3	7.9	**43.3**	13.4	8.9	87.3	16.2	10.0	90.5	18.0
SSE [173] *ICCV'15*	8.5	46.9	14.4	0.2	78.9	0.4	2.1	36.4	4.0	7.0	80.5	12.9	8.1	82.5	14.8
LATEM [162] *CVPR'16*	15.2	57.3	24.0	0.1	73.0	0.2	14.7	28.8	19.5	7.3	71.7	13.3	11.5	77.3	20.0
DEVISE [51] *NIPS'13*	23.8	53.0	32.8	4.9	76.9	9.2	16.9	27.4	20.9	13.4	68.7	22.4	17.1	74.7	27.8
ESZSL [132] *ICML'15*	12.6	63.8	21.0	2.4	70.1	4.6	11.0	27.9	15.8	6.6	75.6	12.1	5.9	77.8	11.0
SJE [7] *CVPR'15*	23.5	59.2	33.6	3.7	55.7	6.9	14.7	30.5	19.8	11.3	74.6	19.6	8.0	73.9	14.4
ALE [**?**] *PAMI'15*	23.7	62.8	34.4	4.6	73.7	8.7	21.8	33.1	26.3	16.8	76.1	27.5	14.0	81.8	23.9
WGAN [164] *CVPR'18*	43.7	57.7	49.7	-	-	-	**42.6**	36.6	**39.4**	**57.9**	61.4	59.6	-	-	-
RG-GZSL [11] *CVPR'18*	31.9	-	40.0	-	-	-	32.9	-	25.5	42.2	-	54.1	**56.7**	-	**65.1**
MLZSL	22.9	60.0	33.2	3.4	77.9	6.5	19.6	45.0	27.3	16.5	78.6	27.3	15.8	79.2	26.3
AE	16.3	31.6	21.5	24.8	38.9	29.6	9.1	22.0	12.9	49.3	52.8	51.0	49.5	63.3	55.7
Ad. AE	17.3	51.9	25.9	28.0	60.4	38.2	11.2	27.1	15.8	56.6	67.0	**61.3**	50.8	76.6	61.0
C-WGAN	47.4	56.1	51.4	27.6	65.9	38.9	40.3	37.2	38.7	54.8	65.7	59.8	49.1	76.2	59.7
GMMN	**49.1**	55.9	**52.3**	**28.5**	64.4	**39.5**	39.7	37.7	38.7	51.5	70.1	59.3	46.3	77.3	57.9

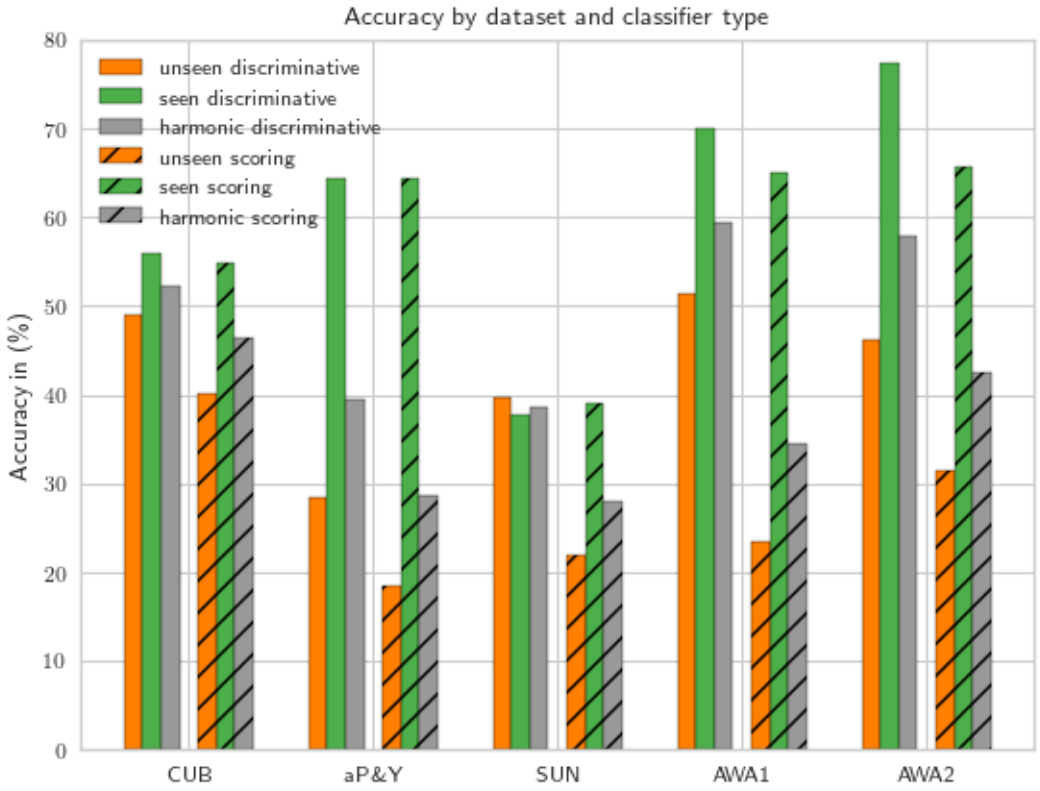


FIGURE 2.10 – GMMN - classifieur discriminant vs fonction de similarité sur la tâche de classification zero-shot généralisée

Deux travaux très récents [11, 164] (publiés à CVPR'2018) qui décrivent des approches très similaires à celle que nous avons précédemment décrite dans [22], avec l'ajout d'une contrainte de classification sur les caractéristiques générées, confirment l'avantage des données générées pour résoudre le problème de classification zero-shot.

Le biais dans la fonction de décision mentionné dans la section 2.1 pour les approches par plongement sémantique peut être constaté dans le tableau 2.4 (colonne u), où la précision diminue significativement par rapport aux performances zero-shot habituelles. Par exemple, la performance de notre modèle MLZSL sur l'ensemble de données AWA2 passe de 62.2% en classification zero-shot à 15.8% en généralisée. Une des raisons de cette baisse est que la distribution des ensembles de données zero-shot est fortement sujette à ce biais, car les classes non vues sont très simi-

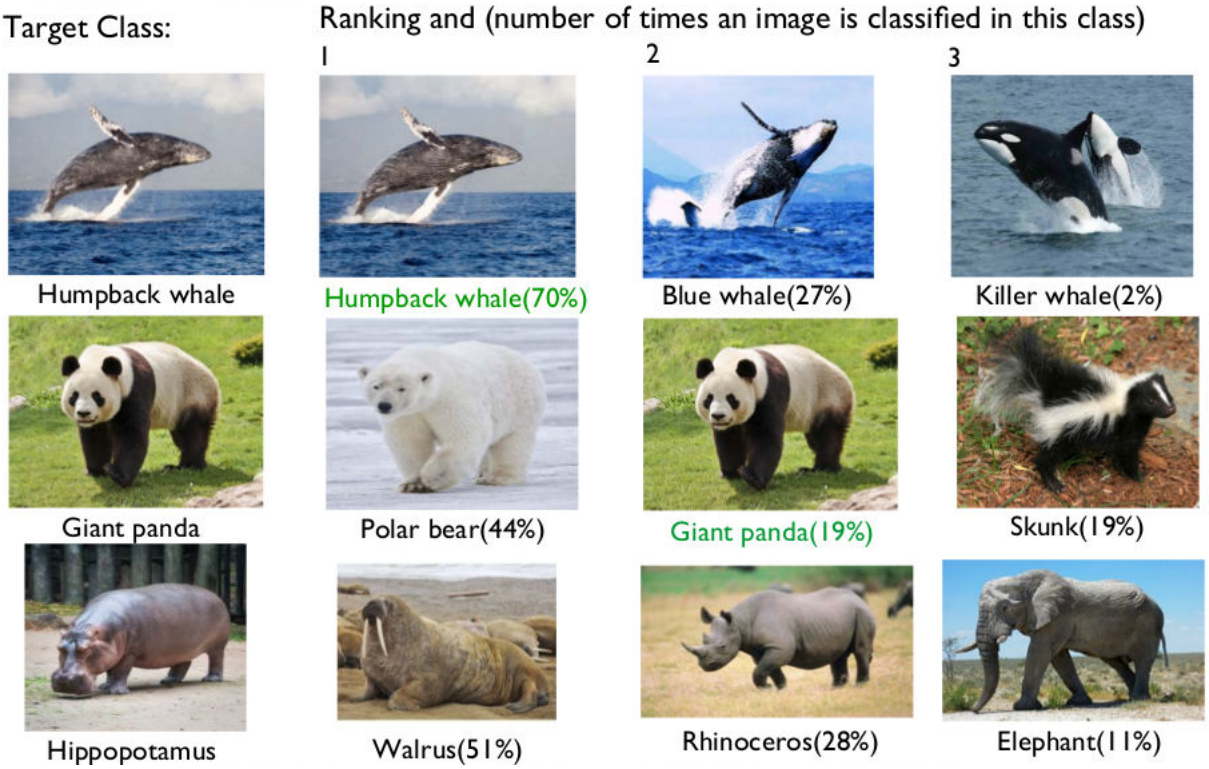


FIGURE 2.11 – Résultats qualitatifs pour le modèle GMMN. La proximité entre les représentations d'attributs influe sur la capacité de discrimination du modèle.

laires aux classes vues à en termes d'apparence visuelle et de description des attributs (ex. zèbre vs cheval). De plus contrairement aux approches génératives, les méthodes par plongement non pas accès aux données des classes non vues pour l'apprentissage de la fonction de décision, il est alors beaucoup plus difficile de les distinguer. Un effet constaté empiriquement, la performance moyenne de notre modèle MLZSL augmente de 12.7% (24.1% vs 36.8%) quand les caractéristiques générées des classes non vues sont disponibles pendant l'apprentissage de la métrique. À noter également que comme pour la tâche de classification zero-shot, l'utilisation d'un classifieur discriminant avec les caractéristiques visuelles permet une hausse des performances de plus de 9.0% en moyenne (voir figure 2.10).

Limite de la représentation par attribut La figure 2.11 montre les classes les plus ambiguës identifiées par notre modèle GMMN sur l'ensemble de données AWA2. En plus d'être visuellement très similaires, la représentation sémantique de certaines classes ne diffère que de quelques attributs. Par exemple, le panda géant et l'ours blanc n'ont que deux attributs différents (couleur de la fourrure et type d'habitat), situation similaire pour la baleine à bosse et la baleine bleue. Les erreurs peuvent donc être produites non pas par un défaut de qualité du classifieur mais par le manque de richesse de la représentation sémantique.

2.4.5 Classification zero-shot à grande échelle

Nous comparons notre approche avec des méthodes de l'état de l'art sur la tâche de classification zero-shot à grande échelle. Cette expérience reprend celle présentée dans l'article [51] : 1000 classes, celles de l'ensemble 1K ImageNet 2012 [39], sont choisies pour l'apprentissage (classes vues) alors que les 20.345 autres sont considérées comme des classes non vues. Les caractéristiques image sont calculées avec le réseau convolutif GoogLeNet [148]. Les catégories sont représentées en utilisant un modèle de langage skip-gram [110] (voir section 2.4.1).

Comme dans [29, 51] notre modèle est évalué sur trois scénarios différents, avec un nombre croissant de classes non vues : i) 2-hop : 1509 classes ii) 3-hop : 7.678 classes, iii) Tous : toutes catégories. Nous utilisons la métrique Flat-Hit @ K qui correspond au pourcentage d'images de test pour lequel le modèle renvoie les vraies étiquettes parmi les K premières prédictions.

La table 2.5 reporte les performances sur les 3 sous-ensembles. Notre modèle génératif GMMN obtient des performances à l'état de l'art pour toutes les configurations. Mais celles-ci restent faibles, la principale explication provient de la nature de la représentation sémantique utilisée pour décrire les classes d'objet. Bien que les approches «word2vec» aient démon-

TABLE 2.5 – Classification zero-shot à grande échelle sur l'ensemble de données ImageNet. Nous reportons la précision moyenne pour différents scénarios (sous ensembles et généralisé). Les caractéristiques visuelles sont extraites d'un réseau GoogLeNet [148].

			Flat Hit @K			
Scénario	**Méthode**	**1**	**2**	**5**	**10**	**20**
2-hop	Frome [51] *NIPS'13*	6.0	10.0	18.1	26.4	36.4
	Norouzi [117] *ICLR'13*	9.4	15.1	24.7	32.7	41.8
	Changpinyo [29] *CVPR'16*	10.5	16.7	28.6	40.1	52.0
	GMMN	**13.1**	**21.5**	**33.7**	**43.9**	**57.3**
2-hop (+1K)	Frome [51] *NIPS'13*	0.8	2.7	7.9	14.2	22.7
	Norouzi [117] *ICLR'13*	0.3	7.1	17.2	24.9	33.5
	GMMN	**4.9**	**13.0**	**20.8**	**31.5**	**45.3**
3-hop	Frome [51] *NIPS'13*	1.7	2.9	5.3	8.2	12.5
	Norouzi [117] *ICLR'13*	2.7	4.4	7.8	11.5	16.1
	Changpinyo [29] *CVPR'16*	2.9	4.9	9.2	14.2	20.9
	GMMN	**3.6**	**6.0**	**11.0**	**16.5**	**23.9**
3-hop (+1K)	Frome [51] *NIPS'13*	0.5	1.4	3.4	5.9	9.7
	Norouzi [117] *ICLR'13*	0.2	2.4	5.9	9.7	14.3
	GMMN	**2.0**	**4.0**	**6.7**	**11.7**	**16.3**
All	Frome [51] *NIPS'13*	0.8	1.4	2.5	3.9	6.0
	Norouzi [117] *ICLR'13*	1.4	2.2	3.9	5.8	8.3
	Changpinyo [29] *CVPR'16*	1.5	2.4	4.5	7.1	10.9
	GMMN	**1.9**	**3.0**	**5.7**	**8.3**	**13.1**
All (+1K)	Frome [51] *NIPS'13*	0.3	0.8	1.9	3.2	5.3
	Norouzi [117] *ICLR'13*	0.2	1.2	3.0	5.0	7.5
	GMMN	**1.0**	**1.9**	**5.0**	**6.2**	**10.3**

tré leur capacité à encoder la sémantique d'un mot, leur performance de discrimination pour une tâche de classification d'image n'est pas garantie. En effet aucune information visuelle explicite n'est fournie lors de l'apprentissage, la fonction de coût prenant uniquement en compte le contexte textuel. De plus l'ensemble de données ImageNet est composé d'un certain nombre de classes d'objets très similaires (ex. de nombreuses races de chien). C'est classes ont de grande chance de partager un contexte textuel semblable, alors que visuellement différentes, rendant leur distinction très difficile.

2.5 Conclusion

Ce chapitre introduit une nouvelle façon d'aborder les tâches de classification zero-shot standard et généralisée. Contrairement aux approches traditionnelles de la littérature, nos travaux se basent sur l'apprentissage d'un générateur conditionnel capable de générer des exemples d'entraînement artificiels pour les catégories non vues. Une fois générés, les exemples sont utilisés conjointement avec l'ensemble de données pour l'apprentissage d'un classifieur discriminant. Notre approche transforme la tâche de classification zero-shot en un problème d'apprentissage supervisé.

Nous avons proposé et comparé quatre architectures différentes de générateurs, le GMMN a permis d'obtenir les meilleures performances de reconnaissance visuelle.

Cette nouvelle formulation aborde les deux principales limites des méthodes zero-shot précédentes : le biais pour les classes vues dans la fonction de décision pour la tâche de classification zero-shot généralisée ; l'utilisation d'un classifieur discriminant dans l'espace des caractéristiques visuelles pour supprimer le «hubness problem». L'utilisation d'exemples artificiels pour représenter classes non vues pendant l'apprentissage du classifieur permet de réduire le biais pour les classes vues. De plus notre approche ne souffre pas du «hubness problem», inhérent aux approches par similarité.

Les expériences réalisées sur six ensembles de données valident expérimentalement la méthode et donnent des performances supérieures à l'état de l'art. Cependant les résultats de reconnaissance visuelle sur l'ensemble de données ImageNet restent faibles, la principale explication provient de la nature de la représentation sémantique utilisée pour décrire les classes d'objets. Dans un objectif de mise en pratique de ce type de technique, une nouvelle façon de générer les représentations sémantiques est indispensable.

CHAPITRE 3

GOULOT D'ÉTRANGLEMENT SÉMANTIQUE POUR LA DÉTECTION DE DÉFAILLANCE

Ce chapitre introduit une nouvelle méthode de représentation de l'image qui est par nature sémantique, abordant la question de l'intelligibilité des calculs pour les tâches de vision par ordinateur. Plus précisément, notre proposition est d'introduire ce que nous appelons un «goulot sémantique» dans le processus de traitement. La représentation de l'image est exprimée entièrement en langage naturel, tout en conservant l'efficacité des représentations numériques.

Nous montrons que notre approche est capable de générer des représentations sémantiques qui donnent des résultats au niveau de l'état de l'art pour une tâche de recherche d'images et qui fonctionnent également très bien pour la tâche de classification multi-étiquettes. L'intelligibilité est évaluée à partir de la capacité de l'utilisateur à détecter des défaillances de prédiction.

3.1 Introduction

L'avènement de l'apprentissage profond en vision par ordinateur a permis l'amélioration des performances empiriques. Les caractéristiques profondes et les processus d'inférence prédictive sont notoirement difficiles à interpréter : les étapes requises pour produire les prédictions sont trop complexes ou nombreuses pour être comprises par les humains. Cette opacité limite leur utilisation pour des applications critiques, comme le diagnostic médical par exemple, qui nécessite des éléments compréhensibles pour la sécurité des étapes de décision et pour convaincre les autorités légales ou les praticiens que le traitement peut être utilisé avec confiance.

La recherche de représentations d'images efficaces et polyvalentes a été l'une des tendances importantes et cruciales des travaux en vision par ordinateur depuis ses débuts. Avec l'avènement des techniques de réseau profond pour la compréhension de l'image, elle a été de facto rétrogradée à un rang secondaire dans les programmes de recherche.

Les tâches reliant image et texte (VQA, captioning ...) ont fait d'énormes progrès depuis l'avènement des approches d'apprentissage profond (voir [37]). Le travail présenté dans ce chapitre s'appuie sur ces nouvelles approches et a pour objectif d'évaluer la capacité des représentations textuelles à encoder sémantiquement et complètement le contenu visuel des images. Nous forçons la fonction de prédiction à passer par un goulot d'étranglement sémantique : étape intermédiaire du processus de décision entièrement textuelle (voir figure 3.1). Le développement récent de nouvelles tâches de vision par ordinateur telles que le «captioning» ou le «visual question answering» (VQA) rend possible la construction de représentations sémantiques plus expressives. Notre proposition pour générer la représentation sémantique consiste à associer une description d'image textuelle et un quiz visuel sous la forme d'une petite liste de questions et de réponses.

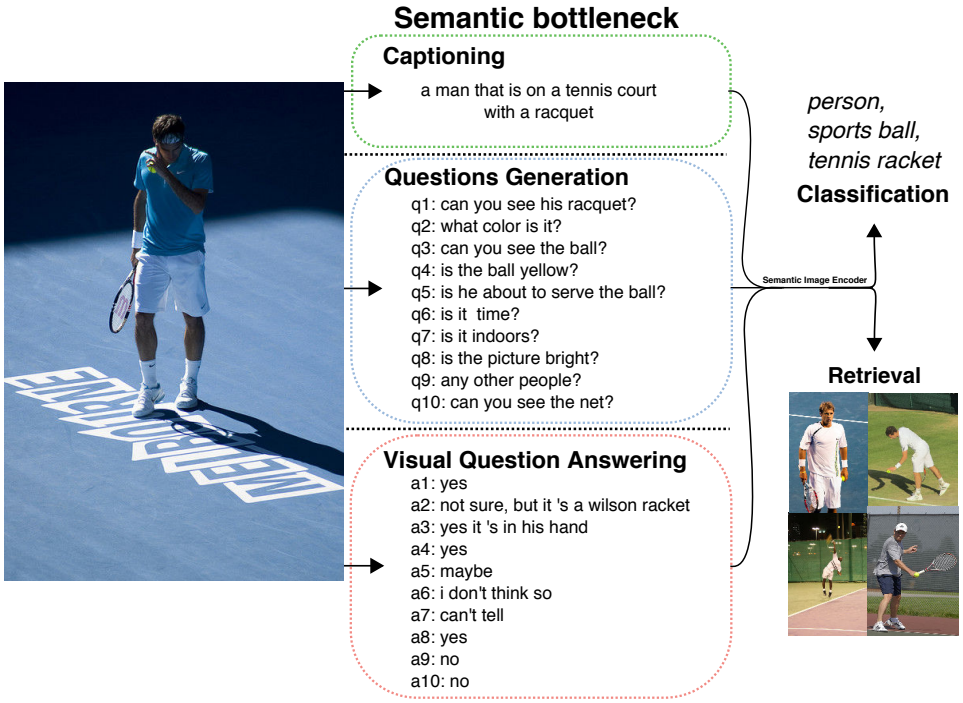


FIGURE 3.1 – Goulot d'étranglement sémantique : les images sont remplacées par une représentation textuelle riche (description et quiz visuel), pour des tâches telles que la classification multi-étiquettes ou la recherche d'images.

L'objectif principal d'un goulot d'étranglement sémantique est de jouer le rôle d'une explication du processus de prédiction, car il offre l'opportunité d'examiner de manière significative sur quelles informations les prédictions seront faites, et de décider éventuellement de les rejeter. Un tel goulot d'étranglement sémantique explicable devrait être un bon compromis entre la précision de la prédiction et l'interprétabilité [56].

Évaluer de manière fiable la qualité d'une explication n'est pas simple [21, 41, 56, 128]. Un objectif possible est d'exploiter ce type de représentation, soit comme une sortie intermédiaire pour fournir des informations sur le comportement de la chaîne de traitement, soit comme point d'entrée pour un contrôle plus intelligible. Dans ce travail, nous proposons d'évaluer le pouvoir explicatif du goulot d'étranglement sémantique en mesurant sa capacité à détecter l'échec de la fonction de prédiction, soit grâce à un détecteur automatisé comme [170], soit par jugement humain.

Les principales contributions de ce chapitre sont :

(i) La conception de deux chaînes de traitement pour la recherche d'images et la classification multi-étiquettes composées d'un goulot d'étranglement sémantique ;

(ii) Une méthode de sélection séquentielle d'une liste de questions et de réponses pour former un quiz visuel sémantique ;

(iii) Une approche globale de fusion exploitant conjointement les différentes composantes de la représentation sémantique pour la recherche d'images ou la classification multi-étiquettes ;

(iv) Une évaluation complète sur la base de données MS-COCO exploitant les annotations Visual Dialog [37] et montrant qu'il est possible d'imposer un goulot d'étranglement sémantique avec seulement 5% de perte de performance pour la classification multi-étiquettes, et un gain de performance de 10% pour la recherche d'images, par rapport aux approches basées sur les caractéristiques profondes de l'image ;

(v) Une évaluation de la capacité d'explication de goulot d'étranglement sémantique comme un moyen de détecter l'échec dans le processus de prédiction et d'améliorer sa précision par le rejet.

Le reste de ce chapitre est organisé comme suit. La section 3.2 présente l'état de l'art des travaux proches de notre problématique. Puis dans la section 3.3 nous présentons des travaux préliminaires validant la capacité d'une représentation sémantique à encoder le contenu visuel d'une image. Le modèle global est décrit dans 3.4 et évalué dans la section 3.5.

3.2 État de l'art

Le travail proposé dans ce chapitre est lié à plusieurs domaines distincts. Tout d'abord, l'introduction d'un goulot d'étranglement sémantique nécessite de pouvoir produire automatiquement une description sémantique des images. Deuxièmement, l'extraction automatique d'informations sémantiques nécessite souvent de transférer des informations apprises sur des domaines connexes pour lesquels des annotations riches sont disponibles (par exemple, en utilisant Wikipédia, l'ensemble de données VQA, etc.). Troisièmement, notre travail est également lié à la question de savoir comment utiliser des représentations intelligibles pour comparer des images ou rechercher dans des ensembles de données. Cette section couvre ces trois aspects du problème.

3.2.1 Extraction d'informations sémantiques

La représentation d'images par des attributs sémantiques a reçu beaucoup d'attention dans la littérature récente. Notamment en classification zero-shot [24, 84, 132, 155, 162, 174], avec pour objectif de construire un espace sémantique dans lequel les images et les attributs peuvent être directement comparés. Cependant, à l'exception du modèle DAP [84], aux performances de reconnaissance faibles, de tels modèles produisent des représentations vectorielles qui ne sont pas directement intelligibles.

En revanche, les descriptions d'images proposées par [97, 152] produisent, par nature, des représentations intelligibles et peuvent être utilisées pour indexer des images. À titre d'exemple, Gordo et al. [59] propose une tâche consistant à retrouver des images partageant la même sémantique que l'image de requête à l'aide de description textuelle. En dépit du succès de ces méthodes, [36] observe que de telles approches produisent des légendes similaires lorsqu'elles contiennent un objet commun, malgré leurs différences dans d'autres aspects. Pour résoudre ce problème,

FIGURE 3.2 – *Image extraite de [37] :* Visual Dialog, un agent conversationnel dialogue avec un humain à propos du contenu visuel de l'image.

[35] propose une méthode d'apprentissage pour la description d'image encourageant le caractère distinctif, tout en maintenant la qualité globale du texte généré.

Les modèles de VQA offrent une alternative riche en termes de mise en relation de la sémantique et des images. Le VQA [9] est une tâche consistant à répondre à une question posée, sur une image, en langage naturel. Tirer parti de la représentation de l'image en utilisant cette connaissance est une direction intéressante, car cette connaissance combine des informations visuelles et sémantiques, offrant un pont entre les deux domaines.

Une autre façon d'enrichir la représentation sémantique est de générer un ensemble de questions et de réponses tel que proposé dans le cadre d'un quiz visuel [37, 96, 137] (voir figure 3.2). C'est ce que nous proposons de faire en apprenant à construire un dialogue complémentaire à la description textuelle de l'image.

3.2.2 Transfert d'informations entre domaines

Produire une description sémantique d'image en langage naturel est possible grâce au transfert d'informations sémantiques - exprimées en attributs, en langage naturel, dictionnaires, etc. C'est exactement ce que les modèles de VQA peuvent faire, offrant des ressources importantes, sous la forme d'images, de questions et de réponses possibles.

La recherche sur le VQA a été très active au cours des deux dernières années. [103] propose une approche basée sur un réseau neuronal récurrent utilisant une mémoire à long terme (LSTM). Dans leur approche, l'image (caractéristiques CNN) et la question sont introduites dans le LSTM, les réponses sont générées par le réseau récurrent. [93] utilise les connaissances sémantiques d'un modèle VQA pour améliorer la recherche d'images. [176], motivé par l'objectif de développer un modèle basé sur un ensemble de régions de l'image, propose un modèle attentionnel pour effectuer cette tâche. À l'aide d'une décomposition de Tucker, [17] modélise finement les interactions entre l'image et la modalité textuelle, tout en contrôlant la complexité de la relation bilinéaire. De son côté, [72] développe un modèle sous la forme de triplet image-question-réponse, optimisé à prédire si oui ou non le triplet est correct. Enfin, [160] introduit une base de connaissances générales pour rendre possible la réponse à des questions plus complexes.

3.2.3 Produire des représentations intelligibles

La représentation de l'image est l'une des questions fondamentales abordées par la communauté de la vision par ordinateur. La littérature regorge de méthodes pour représenter les images, depuis les représentations basées sur les histogrammes (voir [31, 153]) populaires dans les années 90, les sacs-de-mots-visuels [144], jusqu'aux représentations très récentes par réseaux convolutionnels profonds [133, 143] et les vecteurs

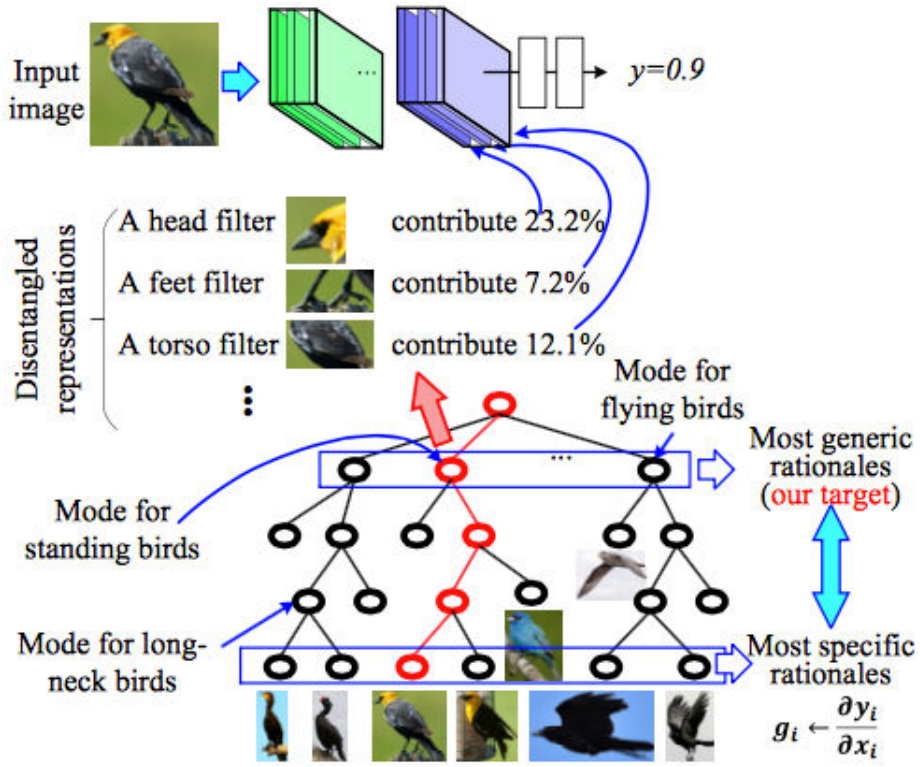


FIGURE 3.3 – *Image extraite de [172] :* étant donné une image d'entrée, l'arbre de décision analyse quelles parties d'objet sont utilisées dans la prise de décision.

de Fisher [82, 126].

L'omniprésence des réseaux neuronaux profonds dans les chaînes de traitement modernes, notamment leur complexité structurelle et leur opacité, ont motivé le besoin d'introduire une certaine intelligibilité dans le processus de prédiction pour mieux comprendre et contrôler leur comportement. Le vocabulaire et les concepts liés aux problèmes d'intelligibilité ne sont pas clairement définis. La littérature parle d'explication, de justification, de transparence, d'interprétabilité, de compréhension, d'introspection, de fiabilité, de responsabilité, d'équité, d'exhaustivité ...

Plusieurs articles récents ont essayé de clarifier ces expressions [21, 40, 41, 56, 61, 70, 94, 128] et séparent les différentes approches en deux buts : construire des modèles interprétables et / ou fournir une justification de la prédiction.

Par exemple, [172] décrit une méthode sous la forme d'un arbre de décision capable d'expliquer la logique de chaque prédiction d'un réseau neuronal convolutif pré-entraîné (voir figure 3.3). La génération d'explications comme justification auxiliaire est abordée sous la forme d'une représentation visuelle de caractéristiques informatives dans l'espace d'entrée, habituellement des cartes de chaleur ou des cartes de saillance [112, 127, 136], comme descriptions textuelles [68] ou les deux [123]. Un grand nombre d'études [42, 101, 119, 169] s'intéressent au rôle des couches ou unités d'un réseau profond par le biais de leur visualisation.

Notre approche de goulot d'étranglement sémantique fusionne ces deux tendances : elle fournit une représentation directement interprétable, qui peut être utilisée comme justification de la prédiction, et force le processus de prédiction à être interprété d'une certaine manière, puisqu'il repose sur une représentation sémantique intermédiaire.

3.2.4 Évaluer une explication

La question de l'évaluation de la qualité ou de la facilité d'utilisation des explications est un problème actif de la littérature. [135] perturbe aléatoirement l'image d'entrée et mesure l'impact sur une carte de chaleur. [136] propose une évaluation expérimentale centrée sur la capacité prédictive d'une explication visuelle. [112] quantifie la qualité de l'explication en mesurant deux caractéristiques : la continuité et la sélectivité des dimensions d'entrée impliquées dans la représentation de l'explication. [171] et [16] décrivent des métriques géométriques pour évaluer la qualité de l'explication visuelle par rapport aux points de repère ou aux objets de l'image. [77] remet en question les explications visuelles basées sur la saillance en montrant qu'un simple changement constant peut conduire à des représentations non interprétables.

Dans notre travail, nous adoptons une approche différente : plutôt que d'évaluer la capacité de l'explication à être utilisée comme substitut ou

FIGURE 3.4 – *Image extraite de [90] :* exemple de paire d'images ambiguës, avec à la fois de bonnes et de mauvaises questions discriminantes.

comme justification d'un processus prédictif, nous évaluons sa capacité à détecter un mauvais comportement.

3.2.5 Générer des questions discriminantes

Si la génération de questions sur des corpus de texte a été largement étudiée [8, 32, 138], la génération de questions sur des images a suscité moins d'attention.

Nous pouvons cependant mentionner le travail intéressant de [90] où des questions discriminantes sont produites pour désambiguïser des paires d'images (voir figure 3.4). Ou [114] qui introduit une nouvelle tâche de génération de questions visuelles. Le travail récent de Das [38] s'appuie sur deux agents communiquant en langage naturel à propos d'une image, les questions et réponses échangées permettent de la retrouver parmi une ensemble de données.

Notre travail corrobore sur les observations faites par [55, 177] :

-les questions posées fournissent des informations sur l'image et

peuvent aider à acquérir des informations pertinentes

-un ensemble de questions discriminantes peut être utilisé pour représenter une image.

3.3 Génération de représentations sémantiques par un système de VQA

Dans cette section nous discutons des travaux préliminaires ayant motivé l'approche proposée dans la section 3.4. Nous souhaitons valider la capacité d'une représentation purement textuelle à encoder le contenu visuel d'une image et à être performante pour les tâches de vision par ordinateur.

L'objectif de la tâche de VQA consiste à répondre à une question posée sur une image, en langage naturel. La plupart des approches actuelles considèrent le processus de réponse comme une classification conditionnelle sous la forme :

$$a = \arg\max_{a' \in \mathscr{A}} L(q, a' \mid I) \text{ for } q \in \mathscr{Q} \tag{3.1}$$

où L est un score de cohérence ou une probabilité mesurant la pertinence de répondre a à une question q à partir de l'ensemble discret de toutes les questions $\mathscr{Q}$ sur une image I. L'ensemble de toutes les réponses possibles $\mathscr{A}$ est également discret et dépend du domaine d'application.

L'idée de cette section est d'utiliser un système VQA comme base de connaissances pour générer la représentation. Ceci peut être réalisé en sélectionnant un sous-ensemble $\mathscr{M} \subset \mathscr{Q} \times \mathscr{A}$ de paires de questions et réponses et évaluer leur probabilité lorsqu'elles sont appliquées à l'image à encoder. La représentation sémantique de l'image S peut être simplement définie comme :

$$S = \{L(q, a \mid I)\}_{(q,a) \in \mathscr{M}} \tag{3.2}$$

Pour chaque paire de question et réponse, nous calculons son score de cohérence. Chaque dimension de la représentation correspond à la probabilité qu'une paire de question et réponse soit compatible avec l'image. Une valeur de vraisemblance élevée est interprétée comme une correspondance entre une image et une paire, comme illustré par la figure 3.5 : la

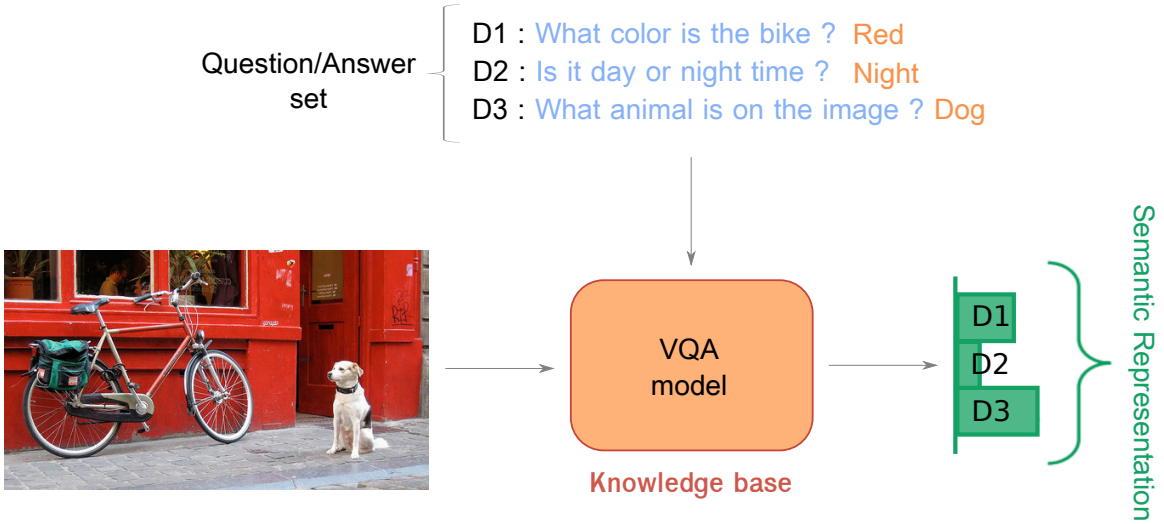


FIGURE 3.5 – Nous proposons d'utiliser un modèle VQA comme base de connaissances pour représenter des images. Chaque dimension de cette représentation correspond à la probabilité qu'une paire question/réponse soit compatible avec l'image.

paire D3 a une probabilité élevée étant donné l'image et correspond à la réponse «dog». contrairement à D2, qui est fausse pour ce triplet.

Pour calculer les vraisemblances des paires, nous utilisons l'approche décrite par [9] : le modèle «deep LSTM Q + norm I» (voir figure 3.6). Les images sont représentées par la dernière couche d'activation d'un réseau VGG-VeryDeep-19 [143], pré-entraîné sur le jeu de données ImageNet [133]. Les questions en langage naturel, représentées sous forme de séquences de mots, sont codées par un LSTM. Afin de réduire la dimensionnalité de la représentation des questions et de correspondre à la taille de l'image, une couche entièrement connectée (suivie d'une fonction de non-linéarité Tanh) réduit la représentation de 2048 à 1024. Les images et les questions sont ensuite combinées grâce à une opération de multiplication élément par élément et appliquées à une couche cachée neuronale entièrement connectée suivie d'une fonction «softmax», produisant la distribution sur les réponses possibles $|\mathscr{A}|$. Le modèle est appris sur l'ensemble de données VQA [9], contenant 204 721 images de MSCOCO [92]. L'ensemble de données comprend environ 760 000 questions avec

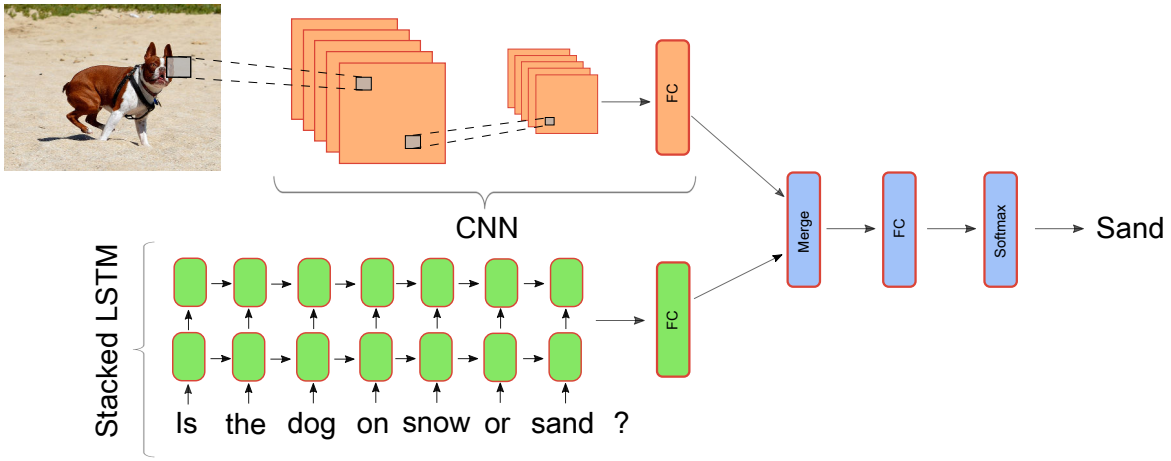


FIGURE 3.6 – «deeper LSTM Q + norm I» : (orange) les images sont représentées par la dernière couche cachée d'un réseau VGG-VeryDeep-19 [143]. (vert) Un LSTM à deux couches code les questions. (bleu) Les représentations de la question et de l'image sont fusionnées via une multiplication élément par élément et utilisées pour la prédiction de la réponse.

presque 10 millions de réponses.

Classification multi-étiquettes Nous évaluons notre approche sur une tâche de classification multi-étiquettes des données MSCOCO. Nous sélectionnons les 10 000 paires les plus fréquentes de l'ensemble de données VQA, afin de pouvoir détecter les régularités statistiques et de réduire le temps de calcul. Après extraction des caractéristiques visuelles ou textuelles nous utilisons un classifieur de type Perceptron pour la classification. La représentation sémantique permet d'obtenir une précision moyenne de 52.8%, performance inférieure de 8.2% par rapport à une approche purement visuelle (61.1%). Ce résultat est encourageant sachant que la représentation utilisée pour la classification est entièrement sémantique.

Sélection des paires L'ensemble des paires de questions et réponses possibles est énorme ($760K \times 10M$ en utilisant l'ensemble de données VQA). Comme l'a montré [72] dans leurs expériences, la distribution des

questions et réponses dans les ensembles de données actuellement disponibles peut être fortement biaisée. Ces faits corroborent l'idée de contrôler soigneusement l'ensemble des paires $\mathscr{M}$ utilisées pour représenter l'image. Dans ces travaux préliminaires, nous avons choisi les 10 000 paires les plus fréquentes, choix arbitraire principalement motivé par le temps de calcul pour la génération de la représentation. Dans la section suivante, nous proposons une méthode automatique de sélection d'un ensemble de questions et réponses complémentaires à une description textuelle de l'image. Nous montrons qu'un faible nombre de paires (10) combinées à une description suffisent pour décrire le contenu visuel de l'image.

3.4 Goulot d'étranglement sémantique

Dans ce chapitre, nous proposons une méthode permettant de transformer des images en représentations sémantiques riches, qui puissent se substituer aux données pour la recherche ou la classification visuelle.

Au cœur de cet encodeur d'image se trouve un processus de génération d'un ensemble ordonné de questions liées au contenu de l'image. Ces questions sont utilisées conjointement avec leurs réponses comme une représentation sémantique de l'image. Les questions sont générées séquentiellement, chaque question (ainsi que sa réponse) induisant les questions suivantes à poser. Un tel ensemble de questions et réponses devrait représenter sémantiquement l'information visuelle des images et être utile pour désambiguïser les représentations d'images dans la tâche de recherche et de classification. Les réponses aux questions sont obtenues avec un modèle VQA, utilisé comme un oracle, jouant le rôle d'un extracteur d'informations. Cette séquence de Q/A est conçue pour être complémentaire d'une description globale d'image pour une tâche donnée. Une analogie avec le raisonnement humain peut être faite, en commençant par la description de l'image comme point de départ et en posant des questions à un oracle pour obtenir itérativement plus d'informations sur l'image. Le dialogue visuel proposé permet d'enrichir la description d'image et conduit à une représentation sémantique plus forte.

La description et le dialogue visuel sont combinés et transformés en une représentation numérique compacte qui peut être utilisée facilement pour comparer des images (pour une tâche de recherche d'images) ou pour déduire des étiquettes de classe (tâches de classification).
Notre modèle est constitué de deux composants :

i) un générateur de questions visuelles discriminantes ;

ii) un bloc de codage prenant le dialogue et la description comme entrées et produisant une représentation numérique.

Ces deux blocs, optimisés conjointement, s'appuient sur deux oracles :

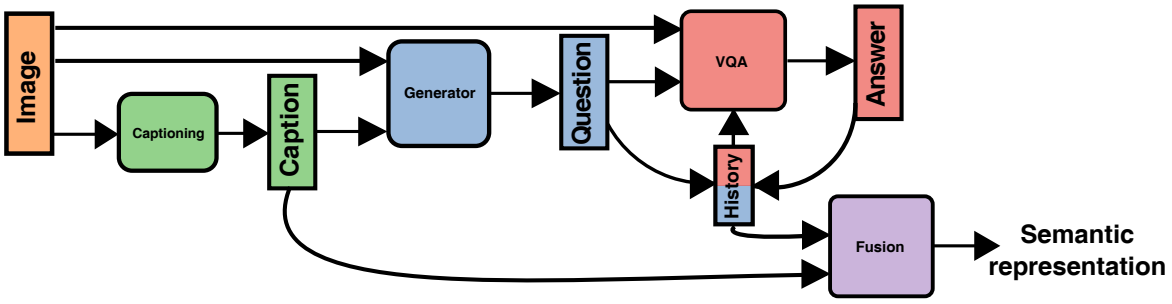


FIGURE 3.7 – Schéma fonctionnel de notre approche.

i) un générateur de description d'image qui décrit visuellement des images avec des phrases en langage naturel ;

ii) un modèle de réponse aux questions visuelles capable de répondre à des questions en rapport avec les images ;

Nous appelons ces deux dernières parties du modèle «oracles» car elles sont obtenues indépendamment des tâches principales et utilisées comme base de connaissances externe (voir figure 3.7).

3.4.1 Représentations vectorielles, encodeurs, décodeurs

L'objectif principal de notre approche est de générer des représentations sémantiques en langage naturel (questions, réponses ou description) qui représentent des images de manière compacte et informative. Nous définissons les différents éléments du langage naturel comme des séquences de mots à partir d'un vocabulaire fixe et de ponctuation $\{w_0, \ldots, w_{n_w}\}$, où w_0 marque la fin de toutes les phrases. L'espace de toutes les séquences possibles en langage naturel est noté $\mathscr{P}$. Toute description (c), question (q) ou réponse (a) appartiennent au même ensemble $\mathscr{P}$.

La plupart des algorithmes basés sur l'apprentissage exploitent des représentations spatiales vectorielles en tant qu'états internes. Un premier problème est donc de rendre possible l'intégration d'expressions du langage naturel, et aussi d'images dans un espace vectoriel. Pour simplifier la conception, nous avons fait en sorte que toutes les représentations

nécessaires appartiennent à un espace vectoriel à valeurs réelles de dimension S. Typiquement dans nos expériences nous avons pris $S = 512$. Nous définissons les «encodeurs sémantiques» comme des projections de $\mathscr{P}$ vers $\mathbb{R}^S$, et les «décodeurs sémantiques» ou «générateurs» comme des projections de l'espace vectoriel $\mathbb{R}^S$ à $\mathscr{P}$.

Dans cette section, nous décrivons les différents processus nécessaires pour intégrer les éléments tels que des images ou les phrases en langage naturel, en représentations spatiales vectorielles.

3.4.1.1 Encodeur d'image

Le premier élément à encoder est l'image $I \in \mathscr{I}$. Le codeur (noté f_I) est une projection $f_I : \mathscr{I} \to \mathbb{R}^S$, fournie par la dernière couche cachée (fc7) d'un réseau VGG-VeryDeep-19 [143] (pré-entraîné sur ImageNet [133]), qui est suivie d'une projection non-linéaire (couche totalement connectée + unité de non-linéarité leakly relu [98]) réduisant la dimensionnalité de la représentation fc7 (4096) à S.

Les paramètres de la projection non-linéaire, notés w_I, sont les seuls paramètres de cette projection qui doivent être appris, les paramètres du VGG-VeryDeep-19 étant considérés comme fixes. Nous écrivons $f_I(I; \mathbf{W_I})$ pour rendre apparente la dépendance sur les paramètres $\mathbf{W_I}$.

3.4.1.2 Encodeur de langage naturel

Cet encodeur projette tout ensemble de mots et de ponctuation $s \in \mathscr{P}$ (descriptions, questions, réponses) dans l'espace d'inclusion $\mathbb{R}^S$. Nous utilisons 3 encodeurs de langage naturel différents dans l'algorithme proposé, pour les descriptions, les questions et les réponses, tous partageant la même structure et les mêmes poids. Nous nous référons donc à eux en utilisant la même notation (f_p). L'encodeur utilise un réseau récurrent à mémoire «long short-term memory» (LSTM) [69], défini par les équations

itératives :

$$
\begin{aligned}
\mathbf{i_t} &= \sigma(\mathbf{W_{ip}}\mathbf{p_t} + \mathbf{W_{im}}\mathbf{m_{t-1}}) \\
\mathbf{f_t} &= \sigma(\mathbf{W_{fp}}\mathbf{p_t} + \mathbf{W_{fm}}\mathbf{m_{t-1}}) \\
\mathbf{o_t} &= \sigma(\mathbf{W_{ip}}\mathbf{p_t}\mathbf{W_{op}}\mathbf{p_t} + \mathbf{W_{ip}}\mathbf{p_t}\mathbf{W_{om}}\mathbf{m_{t-1}}) \\
\mathbf{c_t} &= \mathbf{f_t} \odot \mathbf{c_{t-1}} + \mathbf{i_t} \odot \mathbf{h}(\mathbf{W_{cp}}\mathbf{p_t} + \mathbf{W_{cm}}\mathbf{m_{t-1}}) \\
\mathbf{m_t} &= \mathbf{o_t} \odot \mathbf{c_t}
\end{aligned}
\tag{3.3}
$$

où $\mathbf{c_t}$ est la mémoire à long terme et $\mathbf{m_t}$ celle à court terme, et toutes les variables $\mathbf{W}$ sont des matrices de poids.

Utilisé comme encodeur, le LSTM est simplement un modèle dynamique du premier ordre :

$$\mathbf{y_t} = \mathrm{LSTM}_p(\mathbf{y_{t-1}}, \mathbf{p_t}) \tag{3.4}$$

où $\mathbf{y_t}$ est la concaténation des deux mémoires, à long terme et à court terme, et $\mathbf{p_t}$ est l'entrée courante à l'instant t. Étant donné une séquence de mots en langage naturel $\mathbf{p} = \{\mathbf{p_t}\}_{t=1:T_p}$, son codage $f_p(\mathbf{p})$ est égal à l'état de la mémoire $\mathbf{y_t}$ après avoir itéré sur le LSTM T_p fois, sachant le mot $\mathbf{p_t}$ à chaque itération.

En dénotant $\mathbf{W_p}$ l'ensemble des poids du modèle LSTM, le codage du langage naturel est donc défini comme :

$$f_p(\mathbf{p}; \mathbf{W_p}) = \mathbf{y_{T_p}} \tag{3.5}$$

avec les deux mémoires initialisées à $\mathbf{y_0} = 0$.

En pratique, au lieu de représenter des mots par leur index dans un grand dictionnaire, nous codons les mots dans un espace vectoriel compact de façon similaire au framework word2vec [109]. Cela permet par exemple aux synonymes d'être représentés par des encodages similaires. Plus précisément, ce codage est réalisé sous la forme d'une projection linéaire $\mathbf{W}_{\mathrm{w2vec}}$, où $\mathbf{W}_{\mathrm{w2vec}}$ est une matrice de taille $n_{w2vec} \times n_{\mathrm{w}}$. Chaque mot est encodé en un vecteur de taille n_{w2vec}, n_{w} correspond à la taille

du vocabulaire. La taille de l'espace vectoriel n_{w2vec} est de 200 pour nos expériences. Cette représentation de mots substitue $\mathbf{p_t}$ par $\mathbf{W}_{\text{w2vec}}.\mathbf{p_t}$ en entrée du LSTM.

3.4.1.3 Décodeur sémantique

Le décodeur sémantique est responsable, à partir d'un vecteur $\mathbf{s} \in \mathbb{R}^S$, de produire une séquence de mots en langage naturel appartenant à $\mathscr{P}$. Il est noté f_s.

Nous utilisons 3 décodeurs dans notre approche : un dans le modèle VQA, pour la génération de réponses. Un autre pour produire des questions liées à l'image. Un troisième pour la génération de descriptions. Tous les décodeurs sémantiques que nous exploitons ont la structure d'un réseau LSTM. Ces LSTM ont une sortie prédisant des index de mots selon une couche de classification softmax :

$$\mathbf{p_{t+1}} = \text{Softmax}(\mathbf{y_t}) \tag{3.6}$$

La sortie est réinjectée en entrée du LSTM à chaque itération.

Formellement, nous pouvons écrire le décodeur sémantique comme une suite de mots générés par un processus dynamique de premier ordre avec les observations $\mathbf{p_t}$ comme :

$$\mathbf{y_t} = \text{LSTM}_s(\mathbf{y_{t-1}}, \mathbf{p_t}) \tag{3.7}$$

A l'instant t, l'entrée reçoit le mot généré à l'état précédent $\mathbf{p_t}$, et prédit le mot suivant de la phrase $\mathbf{p_{t+1}}$. Lorsque le mot d'arrêt w_0 est généré à l'instant T_s, il met fin à la génération. Le décodage global est donc une suite de mots de longueur $T_s - 1$:

$$f_s(\mathbf{s}; \mathbf{W}) = \{\mathbf{p_t}\}_{t=1:T_s-1} \tag{3.8}$$

où l'état initial du LSTM est le vecteur à décoder ($\mathbf{y_0} = \mathbf{s}$) et la première entrée est nulle ($\mathbf{p_0} = 0$). Le symbole $\mathbf{W}$ fait référence aux poids du LSTM et

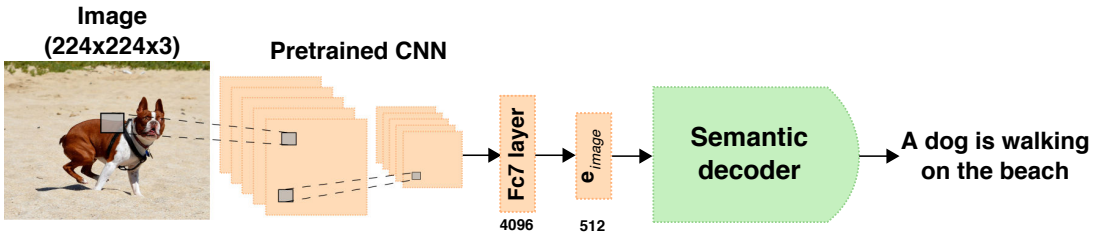


FIGURE 3.8 – Générateur de description : composé de deux parties (1) une extraction des caractéristiques visuelles de l'image ; (2) la génération mot à mot d'une description textuelle.

du softmax, et est différent pour chaque type de donnée textuelle générée (descriptions, questions et réponses).

Nos 3 générateurs ou décodeurs sémantiques (pour descriptions, questions, réponses) ont la même structure, mais des paramètres différents notés respectivement $\mathbf{W}_{\text{cg}}$, $\mathbf{W}_{\text{qg}}$ et $\mathbf{W}_{\text{ag}}$ (en remplaçant la notation générique $\mathbf{W}$ des équations précédentes).

3.4.2 Générateur de description

Le modèle de description visuelle est utilisé comme une source externe de connaissances et est appris en amont. Il prend en entrée une image (I) et produit une phrase en langage naturel la décrivant. Notre approche est inspirée du gagnant du challenge *COCO-2015 Image Captioning* [152].

Elle combine un extracteur de caractéristiques f_I, de type CNN, avec un générateur de phrases f_s, basé sur un réseau récurrent LSTM (voir figure 3.8). Le processus peut être écrit comme :

$$c(I) = f_s(f_I(I;\mathbf{W_I});\mathbf{W_{cg}}) \tag{3.9}$$

où $\mathbf{W_{cg}}$ est l'ensemble de poids appris du générateur. $\mathbf{W_I}$ correspond aux paramètres de la projection non-linéaire, transformant la sortie d'un VGG-VeryDeep-19 en une représentation de taille adaptée au modèle de génération ($S = 512$).

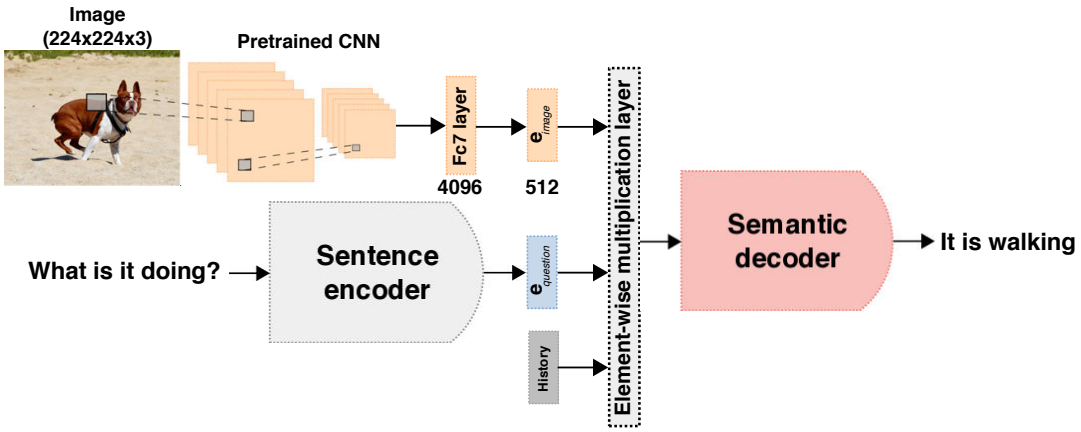


FIGURE 3.9 – Générateur de réponse : à partir de l'image, de la question et de l'historique du quiz visuel, le modèle répond en langage naturel à la question posée.

Pour limiter le coût de génération, nous utilisons une longueur de 1 pour la *«beam search strategy»*, à chaque itération le mot le plus probable est sélectionné. La prédiction de mot est alors donnée par :

$$w_i = \arg\max_{w' \in \mathcal{W}} L_{cpt}(w' \mid w_{0:i-1}, I) \tag{3.10}$$

Ce modèle de description agit comme un oracle, fournissant de l'information sémantique sur l'image pour le module de sélection de questions discriminantes, et pour l'encodeur de représentation sémantique.

3.4.3 Générateur de réponse

Le modèle VQA est le deuxième des deux composants de notre modèle utilisé comme oracle. Il reçoit des questions (Q) en langage naturel et une image (I), et répond en langage naturel à la question sur l'image $a(Q, I)$.

Notre problème est légèrement différent de la tâche de VQA standard, car notre générateur doit maintenant répondre aux questions de façon séquentielle. Cela signifie que la question Q_k peut avoir un lien avec les questions précédentes Q_{k-1} et leurs réponses $a(Q_{k-1}, I)$. Nous présentons d'abord la formulation d'un VQA standard, puis nous montrons comment l'étendre afin de pouvoir répondre aux questions d'un dialogue.

Inspiré par [9], notre modèle VQA combine deux encodeurs, un pour l'image et la question, et un décodeur, pour la réponse (voir figure 3.9). La génération de réponse peut être formulée comme :

$$a(Q,I) = f_s(f_I(I;\mathbf{W_{Iqa}}) \odot f_p(Q;\mathbf{W_{pqa}});\mathbf{W_{qa}}) \tag{3.11}$$

Dans une optique de contrôle de la dimension, la fusion entre l'image et la question est effectuée par un produit terme à terme ($\odot$) entre les deux modalités, comme proposé par [9]. La génération f_s est réalisée suivant l'approche décodeur présentée précédemment.

Nous considérons maintenant le cas où les questions sont extraites d'un dialogue en étendant l'équation 3.11, où k représente la k-ième étape d'un dialogue. Nous introduisons une autre modalité $\mathbf{h_k}$, qui encode les questions et réponses posées du dialogue jusqu'à la k-1 ième étape. La fusion avec l'image et la question est effectuée par un produit terme à terme :

$$a_k(Q_k,I,\mathbf{h_k}) = f_s(f_I(I;\mathbf{W_{Iqa}}) \odot f_p(Q_k;\mathbf{W_{pqa}}) \odot \mathbf{h_k};\mathbf{W_{qa}}) \tag{3.12}$$

L'historique $\mathbf{h_k}$ est simplement calculé comme la moyenne des questions et réponses précédemment posées :

$$\mathbf{h_k} = \frac{1}{k-1}\sum_{i=0}^{k-1}\mathbf{h_i} \tag{3.13}$$

encodé en utilisant f_p. Cet état intègre les questions passées et devrait aider le processus de réponse.

Notre approche de VQA étendu ($\mathbf{W_{Iqa}}$, $\mathbf{W_{pqa}}$ et $\mathbf{W_{qa}}$) est optimisée grâce au jeu de données de Visual dialog [37], composé d'images et de séquences de questions et réponses. La fonction de coût est identique à celle d'un VQA standard. L'erreur est nulle lorsque la réponse donnée est identique à la vérité terrain du dialogue.

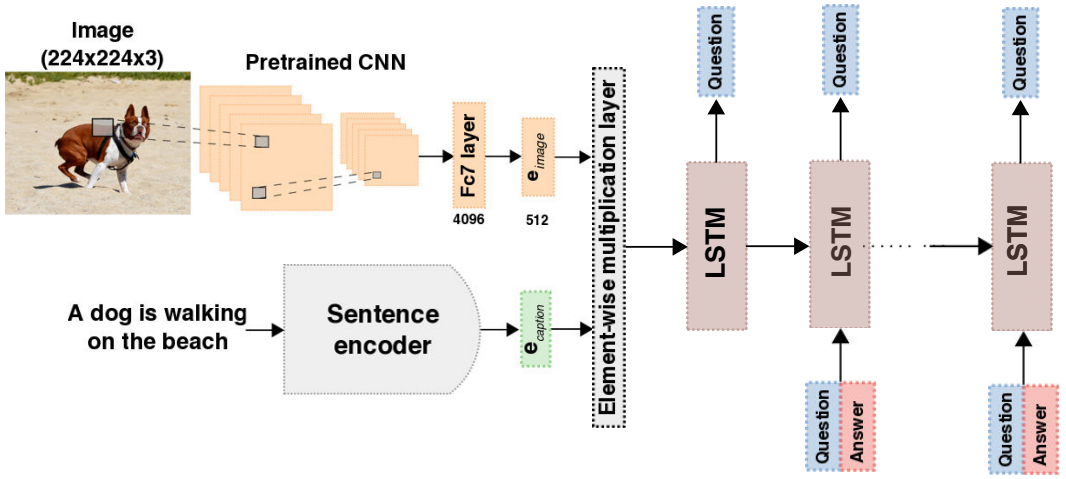


FIGURE 3.10 – Génération de questions discriminantes : un ensemble de questions est produit de manière itérative, dans le but d'affiner la représentation sémantique initiale (description).

3.4.4 Génération de questions discriminantes

Cette partie du modèle produit une séquence de questions et réponses, complémentaires à la description générée (considérée comme la représentation sémantique de base de l'image), pour une tâche spécifique (classification multi-étiquettes ou recherche d'images).

La description de l'image est générée en utilisant le générateur de description présenté dans le Section 3.4.2, notée $c(I) \in \mathscr{P}$ et encodée avec $f_p(c(I); \mathbf{W_p}) \in \mathbb{R}^S$. L'image et la description sont ensuite combinées par un produit terme à terme $f_p(c(I)) \odot f_I(I)$, et utilisées comme représentation initiale de l'image. Cette représentation est ensuite mise à jour de manière itérative en posant et en répondant aux questions, une par une, proposant ainsi une liste de questions discriminantes.

Encore une fois, nous utilisons un réseau LSTM (Fig. 3.10), mais au lieu de fournir un mot à chaque itération, comme dans Eq. (3.8), nous injectons en entrée une paire de question et réponse $[\tilde{\mathbf{q}}_\mathbf{k}, \tilde{\mathbf{a}}_\mathbf{k}]$. $[\tilde{\mathbf{q}}_\mathbf{k}, \tilde{\mathbf{a}}_\mathbf{k}]$ représente la paire $[Q_k, A_k]$ codée dans un espace vectoriel en utilisant Eq. (3.5). La mémoire initiale du générateur de questions est $\mathbf{y_0} = f_I(I) \odot f_p(c(I))$ et

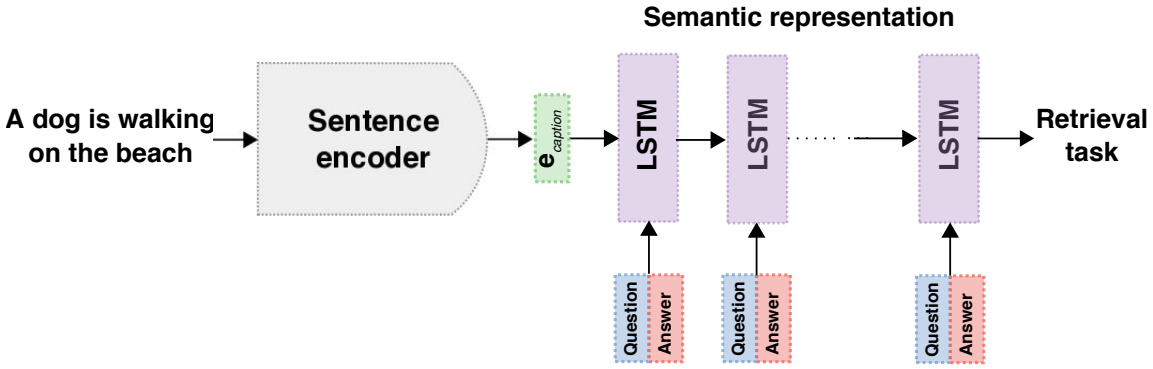


FIGURE 3.11 – Module de fusion : fusionne les représentations sémantiques entre elles (description + quiz visuel). Cette représentation est utilisée pour différentes tâches de vision par ordinateur.

l'entrée initiale est $\tilde{\mathbf{q}}_\mathbf{0} = \tilde{\mathbf{a}}_\mathbf{0} = 0$:

$$\mathbf{y_k} = \text{LSTM}_q(\mathbf{y_{k-1}}, [\tilde{\mathbf{q}}_\mathbf{k}, \tilde{\mathbf{a}}_\mathbf{k}]) \tag{3.14}$$

Les questions sont ensuite décodées à partir de la mémoire interne du LSTM $\mathbf{y_k}$ et transmises au modèle VQA pour obtenir la réponse en utilisant l'équation 3.12 :

$$\begin{aligned} Q_{k+1} &= f_s(\mathbf{y_k}; \mathbf{W_{sq}}) \\ A_{k+1} &= a_{k+1}(Q_{k+1}, I, \mathbf{h_k}) \end{aligned} \tag{3.15}$$

En utilisant ce processus itératif, nous générons, pour chaque image, une séquence de questions et réponses affinant la description initiale :

$$f_q(I; \mathbf{W_q}) = \{Q_k, A_k\}_{k=1:K} \tag{3.16}$$

où $\mathbf{W_q}$ est l'ensemble des poids du réseau et K est un nombre arbitraire de questions.

3.4.5 Encodeur de représentation sémantique

Notre objectif est d'évaluer la faisabilité de substituer une représentation sémantique riche à une image et d'obtenir des performances comparables à une approche basée sur les caractéristiques visuelles pour

plusieurs tâches de vision par ordinateur. Pour être efficace, la représentation doit être spécifiquement générée pour la tâche cible.

Encore une fois, de nombreuses approches modernes de la vision par ordinateur, reposant sur une phase d'apprentissage, exigent que les données soient fournies sous forme de vecteurs de dimension fixe. Le rôle du module décrit ici est de coder la représentation sémantique riche dans $\mathbb{R}^S$ pour alimenter la recherche ou la classification multi-étiquettes.

L'encodeur sémantique utilise un réseau LSTM où la séquence de questions et réponses $\{Q_k, A_k\}_{k=1:K}$ est utilisée comme entrée :

$$\mathbf{y_K} = \mathrm{LSTM}_e(\mathbf{y_{k-1}}, [\tilde{\mathbf{q}}_\mathbf{k}, \tilde{\mathbf{a}}_\mathbf{k}]) \tag{3.17}$$

La description codée $f_p(c(I); \mathbf{W_p})$ est l'état initial de la mémoire $\mathbf{y_0}$ (figure 3.11). La représentation sémantique riche est donc codée comme :

$$f_{\mathrm{enc}}([\{Q_k, A_k\}_{k=1:K}, c(I)]; \mathbf{W_e}) = \mathbf{y_K} \tag{3.18}$$

où $\mathbf{W_e}$ est l'ensemble des poids du LSTM.

3.4.6 Apprentissage du modèle

L'apprentissage est composé de deux phases. La première phase optimise indépendamment les deux oracles : le descripteur d'image et le VQA. La deuxième phase apprend, sur la base des informations fournies par les deux oracles, à générer conjointement le quiz visuel, l'encodeur d'image et le codeur sémantique pour une tâche spécifique.

Les paramètres du modèle VQA, à savoir $\mathbf{W_{Iqa}}$ et $\mathbf{W_{pqa}}$ pour les encodeurs, et $\mathbf{W_{qa}}$ (Eq. 3.12) sont appris de manière supervisée sur le jeu de données Visual Dialog [37]. Le critère d'apprentissage consiste à comparer les mots de réponse générées par le modèle avec ceux de la vérité de terrain par une entropie croisée.

Le générateur de description est également appris en comparant chaque mot généré séquentiellement par l'algorithme à la vérité de terrain, et est également mesuré par un coût d'entropie croisée.

Le générateur de questions et réponses $f_q(I;\mathbf{W_q})$ et le codeur de représentation sémantique de l'équation 3.18 sont appris ensemble de bout en bout. Chacun des modules a sa propre fonction de coût : pour le générateur de questions, la séquence de questions est comparée à la vérité de terrain des questions associées à chaque image en utilisant une entropie croisée. Le codage sémantique est évalué par un coût dépendant de la tâche : un coût d'entropie croisée pour la tâche de classification multi-étiquettes ou un coût sur le rang pour la tâche de recherche d'images.

Lorsque le module de génération de questions converge, seul le coût dépendant de la tâche cible (classification ou recherche) est conservé afin d'affiner la partie de sélection des questions.

Le coût pour la recherche d'images est un peu plus complexe que les autres. Fondamentalement, il est basé sur l'hypothèse que les descriptions de la vérité terrain sont les représentations d'images les plus informatives et que toute autre représentation devrait suivre le même classement de similarité. Nous suivons l'approche proposée par [59] pour définir le coût en fonction de triplets de données q, d^+ (paire positive) et d^- (paire négative) :

$$L(q,d_+,d_-) = \max(0, m - \phi(q)^T\phi(d^+) + \phi(q)^T\phi(d^-)) \tag{3.19}$$

où q et d^+ sont supposées être plus similaires que q et d^-, et ϕ est la fonction de représentation à apprendre (la sortie de l'équation 3.18) et m est un coefficient libre jouant le rôle d'une marge. La similarité entre paires est calculée à partir des descriptions de la vérité terrain en utilisant les représentations *tf-idf*, comme suggéré par [59].

3.5 Expériences

Nous validons la méthode proposée sur 2 tâches :

i) CBIR «Content based image retrieval», tâche de recherche d'images basée sur le contenu sémantique. Nous utilisons le protocole d'évaluation proposé par Gordo [59]. Les descriptions textuelles, encodées avec *tf-idf*, sont utilisées comme vérité terrain. La similarité sémantique de deux images est calculée grâce au produit scalaire de leur représentation *tf-idf*. La performance est mesurée suivant le «Normalized discounted cumulative gain» (NDCG), elle mesure la pertinence des éléments retournés suivant une requête image.

ii) Classification d'image multi-étiquettes : chaque image peut être affectée à différentes classes, indiquant la présence du type d'objet dans la scène. La précision moyenne par classe est l'indicateur de performance de cette tâche.

Les deux séries d'expériences sont effectuées sur le jeu de données Visual Dialog [37], qui utilise les images de l'ensemble de données MS COCO [92]. Chaque image est annotée avec 1 description et 1 dialogue (10 questions et réponses), pour un total de 1,2M questions-réponses. Les dialogues ont été créés dans le but de retrouver une image cible à partir d'un groupe d'images candidates. L'ensemble de questions/réponses décrivent visuellement une image. Nous utilisons le sous-ensemble de données d'entraînement standard pour l'apprentissage et évaluons le modèle sur la partie validation, car l'ensemble de test n'est pas accessible au public.

Notre approche comporte un ensemble d'hyper-paramètres : la taille de la représentation vectorielle de mot, la taille des états internes des $LSTM$, le pas d'apprentissage, la marge m. Ils sont obtenus par validation croisée. 20% des données d'apprentissage sont considérées comme un ensemble de validation, permettant de sélectionner les hyper-paramètres maximisant la précision moyenne / le score NDCG. En pratique, la valeur typique

TABLE 3.1 – Score NDCG pour la tâche de recherche d'images. Performance / Air sous la courbe (AUC) pour différentes valeurs de R.

Méthode / R	8	32	128	AUC
$f_i(I)$ + ML (baseline)	45.8	51.7	59.3	69.7
I + [59]	47.6	55.9	62.3	72.7
$\{I, c(I)\}$ + [59]	57.0	58.5	63.3	75.1
Notre approche $f_{enc}(I)$	**59.3**	**61.7**	**67.1**	**79.9**

TABLE 3.2 – Recherche d'images. Scores NDCG / AUC après avoir supprimé certains composants du modèle.

Modalité / R	8	32	128	AUC
$c(I)$	55.1	56.3	62.4	73.6
$\{Q_k, A_k\}_{1:10}$ générique	41.8	50.4	57.7	65.7
$\{Q_k, A_k\}_{1:10}$ adapté	45.8	55.7	60.0	71.9
$tf-idf\{c(I), \{Q_k, A_k\}_{1:10}\}$	54.9	57.2	63.4	75.1
Notre approche $f_{enc}(I)$	**59.3**	**61.7**	**67.1**	**79.9**

pour la taille des états internes des $LSTM$ (respectivement la taille de la représentation vectorielle de mot) est de 512 (respectivement 200). La marge m est comprise entre [1.0-2.0].

Les paramètres du modèle sont initialisés selon une distribution gaussienne centrée ($\sigma = 0.02$). Ils sont optimisés avec le solveur Adam [78], avec un pas d'apprentissage de 10^{-4} et en utilisant des sous-ensembles de données de taille 128. Afin d'éviter le sur-apprentissage, nous utilisons du dropout [146] pour chaque couche (probabilité de suppression de 0.2 pour les couches d'entrée et de 0.5 pour les couches cachées). Les poids des deux oracles (générateur de description et VQA) sont affinés sur les tâches cibles.

3.5.1 Recherche d'images

Nous évaluons maintenant notre approche sur une tâche de recherche d'images, où les images partageant un contenu sémantique similaire avec l'image cible doivent être retournées par le système.

Comme décrit précédemment, la fonction de coût est optimisée par triplets : une image requête et deux images similaire / dissimilaire. Pour la sélection de triplets, nous appliquons une technique de «hard negative mining» en échantillonnant les images en fonction de la valeur de leur fonction de perte (plus le coût est important plus la probabilité d'être sélectionnée est élevée). Nous avons constaté empiriquement que cette technique est indispensable au bon fonctionnement du modèle.

La table 3.1 montre la performance NDCG pour 3 valeurs de R (où R = k signifie que seules les k premières images sont considérées pour le calcul du NDCG), et l'aire sous la courbe (pour R entre 1 et 256) pour 4 modèles. La baseline exploite une métrique de similarité entre les caractéristiques visuelles extraites de la couche FC7 d'un réseau VGG19. La métrique est optimisée sur l'ensemble d'entraînement en utilisant la même approche par triplet que celle décrite dans la section 3.4.6. I + [59] correspond à l'approche de [59] notée (V, V). $\{I, c(I)\}$ + [59] utilise conjointement l'image et la description textuelle (V + T, V + T), avec la différence que nous n'utilisons pas la vérité terrain, mais des descriptions générées.

L'aire sous la courbe augmente de +4,8% avec notre approche comparée à la baseline. Nous soulignons ici que, contrairement à [59], aucune information visuelle n'est utilisée dans la représentation finale.

Les résultats empiriques dans le tableau 3.2 montrent l'utilité de notre codeur sémantique. En effet, avec les mêmes modalités (descriptions, questions et réponses), $tf-idf\{c(I), \{Q_k, A_k\}_{1:10}\}$ obtient des performances inférieures de 4.8%. La table 3.2 montre également l'importance d'affiner les poids de l'oracle VQA à la tâche cible, avec un gain de +6.2% ($\{Q_k, A_k\}_{1:10}$ générique vs adapté) par rapport à un oracle générique.

TABLE 3.3 – Précision moyennée par classe pour la baseline et les différents composants de notre modèle.

Modalité	mAP
$f_i(I)$ (baseline)	61.1
VQA 10K	52.8
$c(I)$	51.6
$\{Q_k, A_k\}_{1:10}$	49.9
$f_{enc}(I)$	56.0
$\{I, f_{enc}(I)\}$	**64.2**

3.5.2 Classification multi-étiquettes

Chaque image de l'ensemble de données MS COCO [92], est étiquetée avec des labels multi-objets (80 catégories d'objets), représentant la présence de concepts généraux tels qu'un animal, véhicule, personne, etc. dans l'image. Pour l'approche de base (baseline), nous utilisons les caractéristiques visuelles fournies par un réseau VGG-VeryDeep-19 pré-entraîné sur ImageNet [133]. Une couche softmax est apprise sur l'ensemble d'entraînement.

Le tableau 3.3 répertorie la précision moyennée par classe pour la baseline et les différents composants de notre modèle. Notre approche entièrement sémantique $f_{enc}(I)$ souffre d'une baisse de performance de 5% par rapport à l'approche visuelle. C'est assez encourageant, car notre modèle n'utilise qu'une représentation textuelle, la description et les 10 questions / réponses.

La ligne «VQA 10k» correspond aux travaux préliminaires de la section précédente. Bien que les performances de reconnaissance soient légèrement inférieures (49.9 vs 52.8), la sélection automatique de questions permet d'obtenir une représentation de questions et réponses plus compactes (10 vs 10000). De plus le principal avantage de notre approche est que l'on peut avoir accès à la représentation sémantique intermédiaire pour l'inspection, et ainsi fournir une explication du bon ou du mauvais

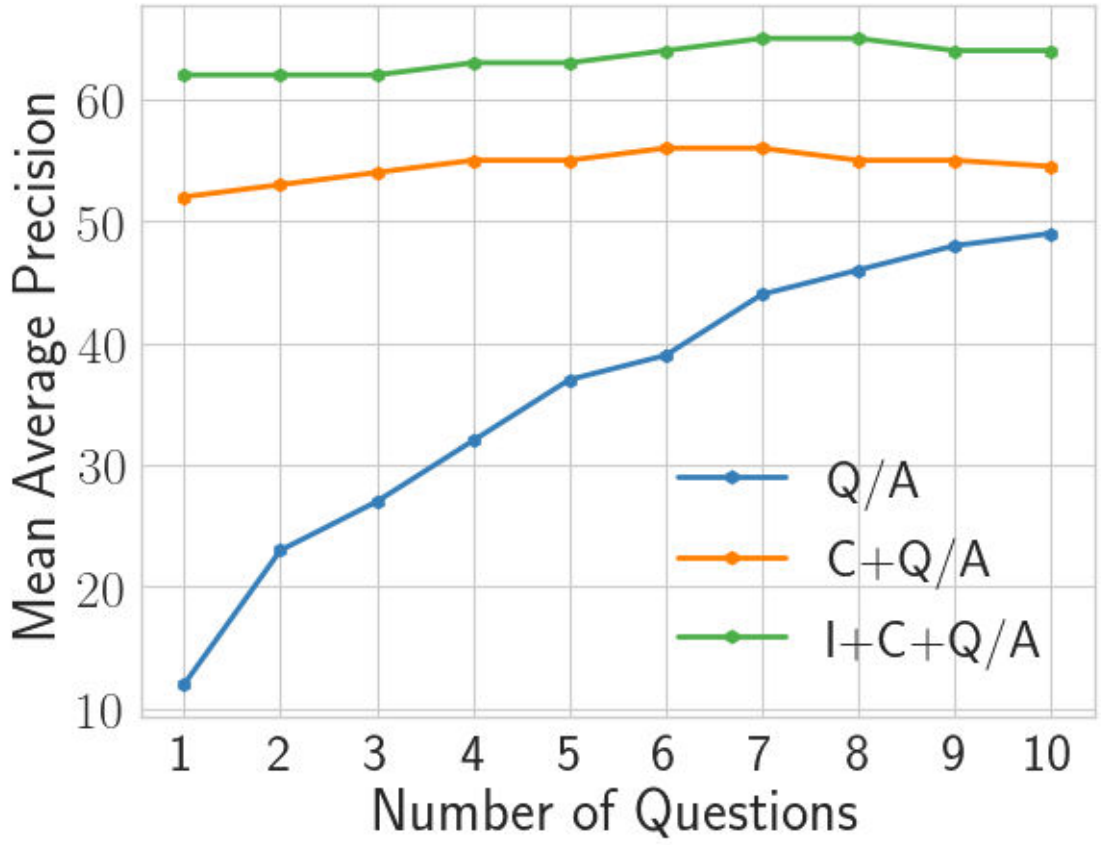


FIGURE 3.12 – Précision en classification multi-labels en fonction du nombre de questions et réponses posées.

fonctionnement (voir la section 3.5.3).

La table 3.3 montre également la performance de classification en utilisant soit la description générée $c(I)$, ou soit les questions et les réponses $\{Q_k, A_k\}_{1:10}$). À partir des résultats d'expériences, nous pouvons conclure que les descriptions sont plus discriminantes que les questions / réponses (+ 1,7 %). La représentation textuelle combinée à celle visuelle (dénotée $\{I, f_{enc}(I)\}$) permet d'obtenir les meilleures performances (+8,2 %) et surpasse la baseline (+3,1 %).

La figure 3.12 montre l'effet du nombre de questions et réponses posées, sur la précision. Quand le dialogue est combiné à la description (C+Q/A et I+C+Q/A) la performance ne s'améliore pas après 7 Q/A. Nous expliquons ce comportement par l'apparition de répétitions dans la liste de questions

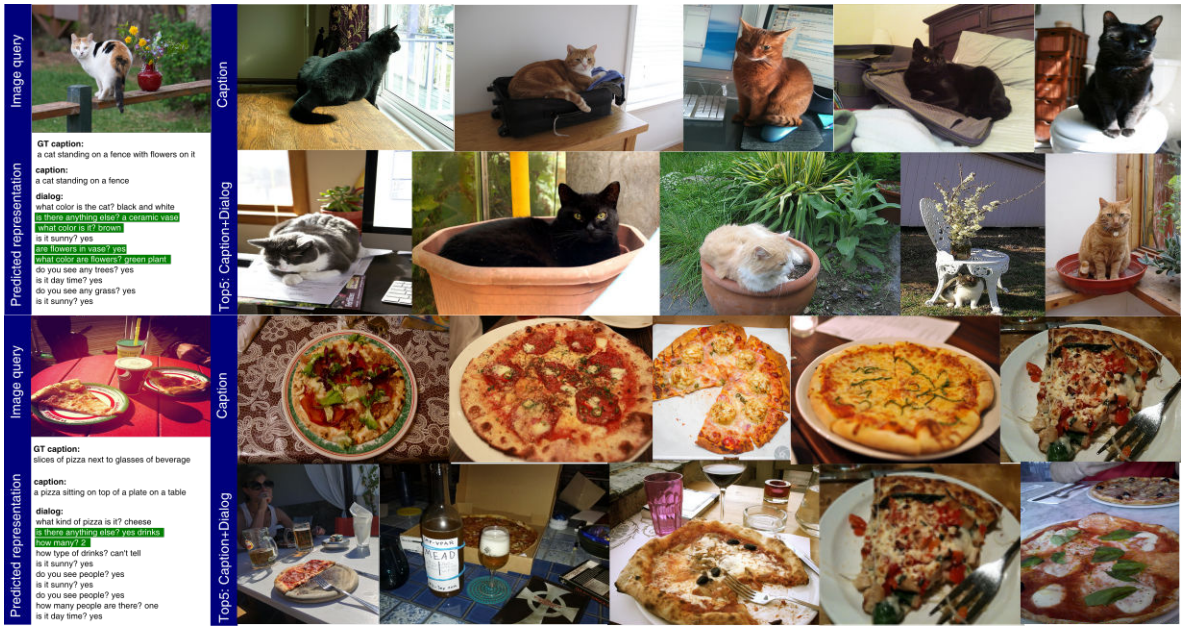


FIGURE 3.13 – Combinaison de descriptions et de dialogues : requête (images en haut à gauche), descriptions générées, dialogues spécifiques à la tâche, images retournées en utilisant la description (premières lignes) et celles données par notre modèle (deuxièmes rangées). Le dialogue permet de détecter des concepts complémentaires importants manqués par la description.

générées (voir section 3.5.3).

En guise de vérification, nous avons également calculé la précision moyenne lorsque nous utilisions des annotations de vérité de terrain pour les descriptions et le dialogue. Nous avons obtenu une performance de 72,2%, ce qui signifie qu'avec de bons oracles, il est possible pour notre goulot d'étranglement sémantique d'obtenir de meilleures performances.

3.5.3 Analyse du goulot d'étranglement sémantique

Cette section vise à donner quelques indications sur i) pourquoi la performance s'améliore en combinant les descriptions et les dialogues et ii) pourquoi adapter le goulot d'étranglement à la tâche améliore la performance.

En ce qui concerne le premier point, nous faisons une analyse qualitative des résultats sur la tâche de recherche d'images, en comparant la pertinence des premières images retournées lors de l'ajout des dialogues aux descriptions. La figure 3.13 est une illustration de ce que nous observons. Pour deux requêtes différentes (images en haut à gauche), la figure montre à la fois la description et le dialogue générés automatiquement, ainsi que les images classées en premier en fonction de la description (premières lignes) et notre modèle combinant la description et le dialogue. Nous avons surligné en vert les informations complémentaires importantes ajoutées par le dialogue. Pour l'image «chat», le dialogue détecte qu'un vase et une plante sont également présents dans l'image. Pour l'image «pizza», le dialogue détecte la présence de boissons. Le dialogue est capable de détecter les boissons comme une caractéristique sémantique importante de l'image.

Concernant le second point, nous comparons la qualité de la recherche avec et sans adapter le dialogue à la tâche. La figure 3.14 illustre nos observations : la figure montre à la fois la description et le dialogue générés automatiquement, ainsi que les images classées en premier en fonction de la description combinée au dialogue générique (premières lignes) et au dialogue adapté à la tâche (deuxièmes rangées). En rouge, les questions et les réponses que nous trouvons non pertinentes pour la requête dans le dialogue générique et en vert celles qui ont été données par le dialogue adapté. Le dialogue a permis d'identifier les véhicules comme une caractéristique importante de l'image. La figure 3.15 illustre le même type de correction de description par le dialogue pour la tâche de classification multi-étiquettes.

De plus, la figure 3.16 montre la différence entre les descriptions, questions et réponses générées pour une tâche de classification ou de recherche. Les descriptions générées sont, en général, brèves et cohérentes avec les images. Les quelques descriptions syntaxiquement incorrectes sont dues à notre stratégie d'échantillonnage (afin d'avoir un compromis

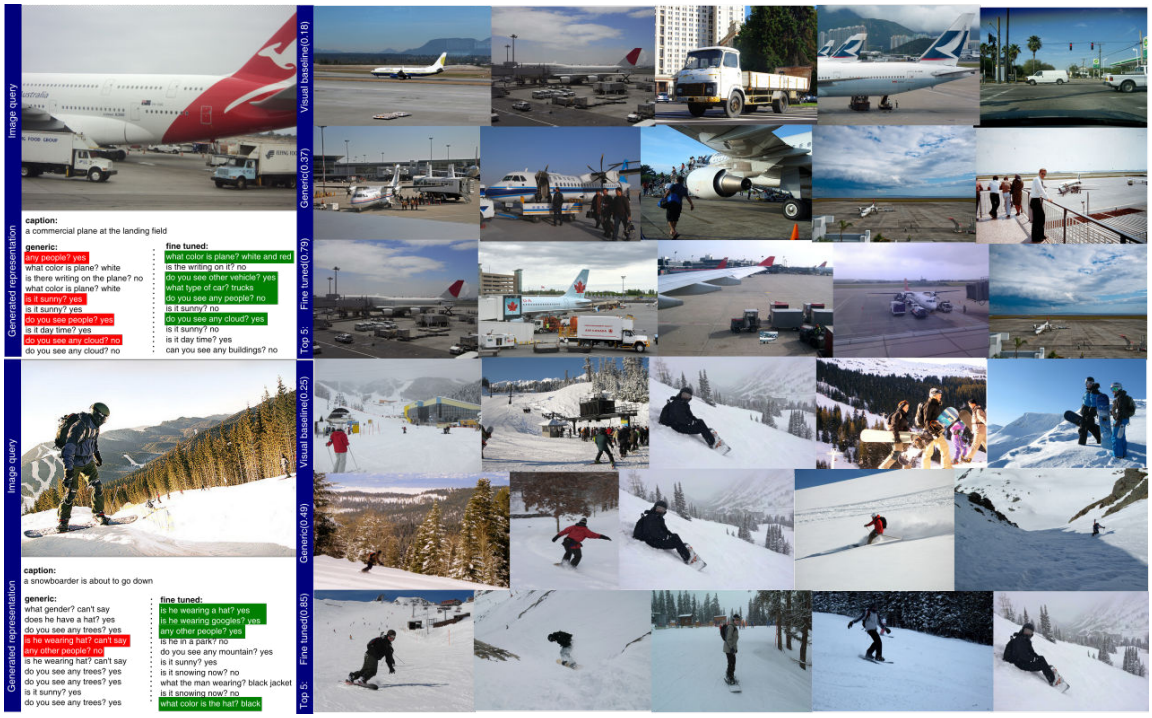


FIGURE 3.14 – Adapter le dialogue à la tâche : requête (images en haut à gauche), légendes générées, dialogue générique et spécifique à la tâche, images récupérées en utilisant la description et le dialogue générique (premières lignes) et celles données par notre modèle (deuxièmes rangées).

Semantic representation:
caption:
people are snowboarding down a mountain
dialog:
how many people are there? two
is there snow? no
where are they snowboarding? a lake
do you see any mountain? no
is it sunny? no cloudy
where are they snowboarding? a lake
is it day time? yes
is there snow? no
do you see any mountain? no
is it day time? yes

Predicted labels: person, skis, boat

Semantic representation:
caption:
a group of men talking
dialog:
how many people are there? can't count
is it sunny? no
is it day time? yes
what are they wearing? a suit
what color? dark blue
is it raining now? yes
is it day time? yes
how many people are there? can't count
is it sunny? no
where are they? on a beach

Predicted labels: person, tie, umbrella

FIGURE 3.15 – Partie gauche : description incorrecte corrigée par le dialogue. Droite : objets manquants dans la description, mais découverts en posant des questions pertinentes.

entre le temps calcul et l'interprétation). Cela ne devrait pas avoir d'impact sur la performance, car les descriptions reportées reflètent le contenu de l'image. De la même manière, on peut voir que plusieurs questions sont répétées. Bien que la répétition des questions ne soit pas aussi critique que pour la génération de dialogues, elle peut être surmontée par exemple

label: person, dog, sheep
generated caption:
a person and a dog are walking next sheep

classification:	**retrieval:**
any people? yes	are people male? female
what kind of animal? sheep	what kind of animal? sheep
is it sunny? yes	how many? 3
is the dog walk? yes	any other animals? dog
is it sunny? yes	is the dog walking? yes
is it sunny? yes	what color is dog? black
what kind of animal? sheep	is it sunny? yes
do you see people? yes	is it sunny? yes
do you see people? yes	what color is dog? black
do you see people? yes	can you see any trees? yes

label: car, stop sign
generated caption:
a car on a street with trees

classification:	**retrieval:**
are there cars? yes	are there cars? yes
any people ? no	how many stop sign? 2
can you see stop sign? yes	any people? no
is it sunny? no	any buildings? yes
any people ? no	do you see any trees? yes
how many car? 1	is it sunny? no
any people? no	any people? no
can you see stop sign? yes	how many car? 1
can you see stop sign? yes	any people? no
are there cars? yes	do you see any trees? yes

label: bowl, chair, couch, potted plant, vase
generated caption:
a living room with coaches and table

classification:	**retrieval:**
any people? no	is it daytime? yes
is it table? yes	is it table? yes
do you see chair? no	any people? no
is this outside? no	do you see chair? no
how many couch? 1	is this photo color? no
is this photo color? no	is this outside? no
any people? no	any couch? yes
is it sunny? no	what is the color? red
is it sunny? no	any people? no
any people? no	is it sunny? no

FIGURE 3.16 – Descriptions et dialogues générés pour les tâches de classification et de recherche.

tâche/mot	classes	playing	eating	wearing	doing	color	how many	in/outdoor	day/night
classification	88%	19%	39%	29%	14%	42%	39%	42%	53%
retrieval	78%	14%	28%	21%	16%	85%	81%	54%	79%

TABLE 3.4 – Pourcentage d'apparition pour les deux tâches. La colonne «classes» correspond au pourcentage d'occurrence de l'ensemble des étiquettes du problème de classification multi-labels.

en pénalisant explicitement les répétitions dans le critère de coût du LSTM ou en exploitant une approche d'apprentissage par renforcement tel que dans [38]. La table 3.4 indique le pourcentage d'apparition des mots dans les représentations sémantiques pour chaque tâche. Les dialogues

Semantic representation:
a man is standing in front of a elephant
how many elephants? 2
are there people in the image? yes
are their trunks in the air? yes
what color is the truck? brown
are there any buildings? no
is it day or night? day
are there any other elephants? no
are there any trees? yes
are there any other animals? no
are there any hills? yes

Predicted labels: person, truck, elephant
Missed labels: -

Semantic representation:
three people are drinking and eating
are they mal or female? female
how many people do you see? 3
how old is the girl? young
what are they eating? a bacan sandwich
is it on table? I can't tell
is it day time? yes
what are they wearing? pants and jackets
is it day time? yes
what are they eating? a bacon sandwich
is it on table? I can't tell

Predicted labels: person, sandwich
Missed labels: pizza, table

Semantic representation:
a group of people standing on top
of a sandy beach
is the photo in color? yes
is it daytime? yes
are there any people? yes
can you see any buildings? no
what color is the kite? white
is it sunny? yes
is this person wearing a hat? no
can you see the water? no
is the sky blue? yes
is the sky blue? yes

Predicted labels: person, kite
Missed labels: boat, umbrella

Semantic representation:
a group of girraffes standing in a field
any people? no
is it grazing? yes
is it grazing? yes
is it grazing? yes
any other animals? no
any people? no
any buildings? no
are there any trees? yes
is it sunny? yes
what are they eating? eat grass

Predicted labels: girraffe
Missed labels: cow

FIGURE 3.17 – Représentations sémantiques défaillantes.

générés à partir de la tâche de classification contiennent davantage de verbes pouvant être associés à la présence d'une classe d'objets (eating ⇒ nourritures, playing ⇒ sport, wearing ⇒ vêtements). Les dialogues générés pour la tâche de recherche d'images contiennent davantage de mots caractérisant globalement la scène (in/outdoor, day/night) ou à une caractéristique spécifique d'un objet (color, how many). Ces différences se remarquent également qualitativement, la figure 3.16 montre l'effet sur la sélection des questions d'adapter les poids suivant la tâche ciblée : les questions générées pour la classification sont liées à la présence de catégories d'étiquettes (voyez-vous une personne ? Un chien ? etc.). Alors que les questions générées pour la recherche d'images sont plus spécifiques (combien ? Quelle couleur ? etc.). La tâche de recherche d'images nécessite des informations plus globales. Nous notons également des questions moins redondantes pour la tâche de recherche, car le modèle a besoin de plus d'informations.

À noter que plusieurs questions et réponses sont fausses (voir figure 3.17), mais nous sommes confiants sur le fait qu'avec l'amélioration générale des modèles, les performances en recherche et de classification d'images augmenteront à l'avenir.

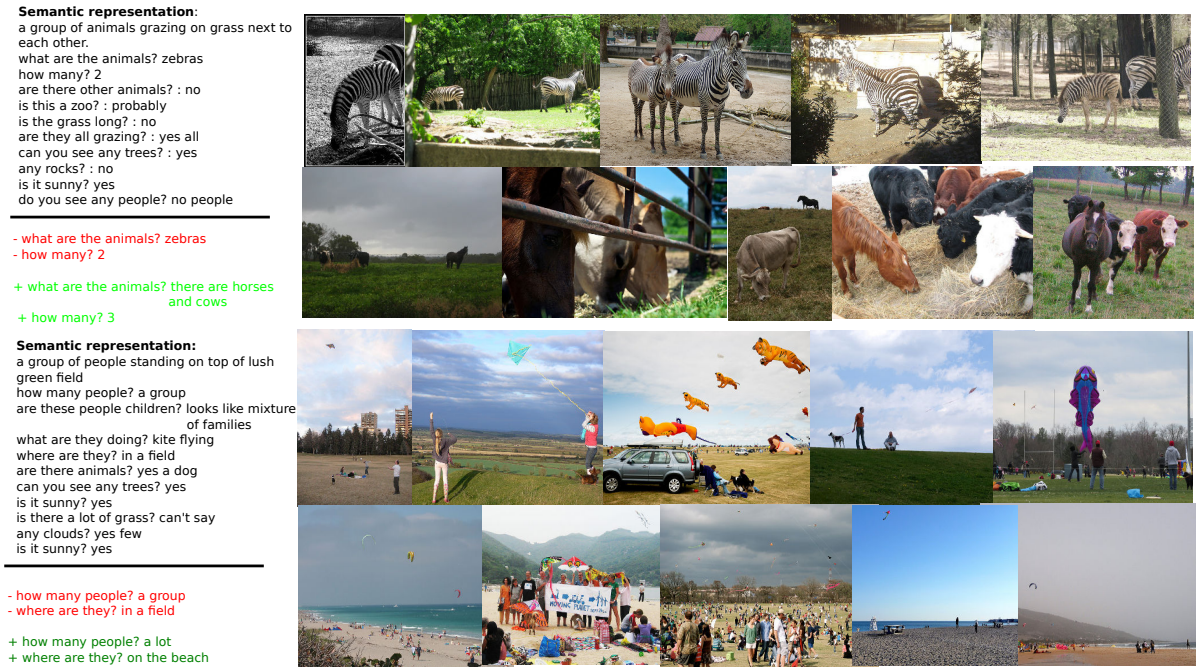


FIGURE 3.18 – Modifications de la représentation sémantique.

3.5.4 Modification de la représentation sémantique

La figure 3.18 montre l'effet sur les résultats de recherche lors de la modification de la représentation sémantique. L'intelligibilité de la représentation permet de former une nouvelle requête en modifiant le dialogue ou la description et ainsi de récupérer des images différentes. Pour la première ligne, nous avons changé 2 réponses (zèbre devient vache et cheval, leur nombre est augmenté de un). Dans la deuxième ligne, le changement d'environnement a bien été pris en compte (de plaine à plage)

3.5.5 Détection de défaillance

La capacité du goulot d'étranglement sémantique à détecter une défaillance dans le processus de prédiction est illustrée par la figure 3.19. Une défaillance est détectée soit lorsque la représentation contient des informations sémantiques incorrectes - la description ou le dialogue sont erronés - ou lorsque l'information extraite est insuffisante pour l'inférence

Semantic representation:
caption:
two cars are crossing each other
dialog:
what color are vehicles? blue and white
do you see any people? no
is it day time? yes
is it sunny? yes
how many car do you see? 2
is it sunny? no
do you see any bulding? no
do you see any tree? yes
how many car do you see? 2
is it day time? yes

Predicted labels: car **Missed labels:** giraffe

Semantic representation:
caption:
a group of people riding surfboards on top of a wave
dialog:
do you see any people? yes
ihow many? 2
is it day time? yes
is it sunny? yes
what clothes? black swimsuits
male or female? male
is it day time? yes
can yousee water? yes
what clothes? black swimsuits
is it day time? yes

Predicted labels: person, surfboard **Missed labels:** dog

FIGURE 3.19 – Prédire les cas d'échec à partir de la représentation sémantique. Côté gauche : la légende et Q/A sont cohérents, mais pas assez riches pour prédire l'étiquette «girafe». Droite : la représentation sémantique est incorrecte, ce qui conduit à l'inférence d'étiquettes erronées. Dans de tels cas, le goulot d'étranglement peut être utilisé pour le débogage.

des classes présentes.

Nous concentrons notre évaluation sur la classification multi-étiquettes d'images, car une définition claire de l'échec dans le cas d'une recherche d'images basée sur le contenu est complexe et peut être subjective (comment décider si les images sont complètement différentes de la demande ?).

Nous avons développé deux protocoles d'évaluation : un avec une interaction humaine, jugeant la capacité du goulot d'étranglement sémantique à prédire les étiquettes cohérentes. Et un modèle automatique optimisé à prédire le succès ou l'échec de chacune des classes. Nous comparons nos approches à une baseline basée sur le seuillage des scores de prédiction.

Afin d'évaluer la capacité du goulot d'étranglement sémantique, le modèle est optimisé pour une tâche de classification multi-étiquettes. Nous extrayons la représentation sémantique générée (description et dialogue) et la prédiction des classes présentes.

Prédiction humaine des défaillances Pour 1000 images de test choisies aléatoirement, plusieurs utilisateurs ont été chargés d'évaluer la capacité du goulot d'étranglement sémantique à contenir suffisamment d'informations pour prédire les classes correctes. L'image et la représentation sémantique générée sont montrées aux utilisateurs, qui peuvent sélectionner (voir interface figure 3.20) pour chacune des 80 étiquettes de

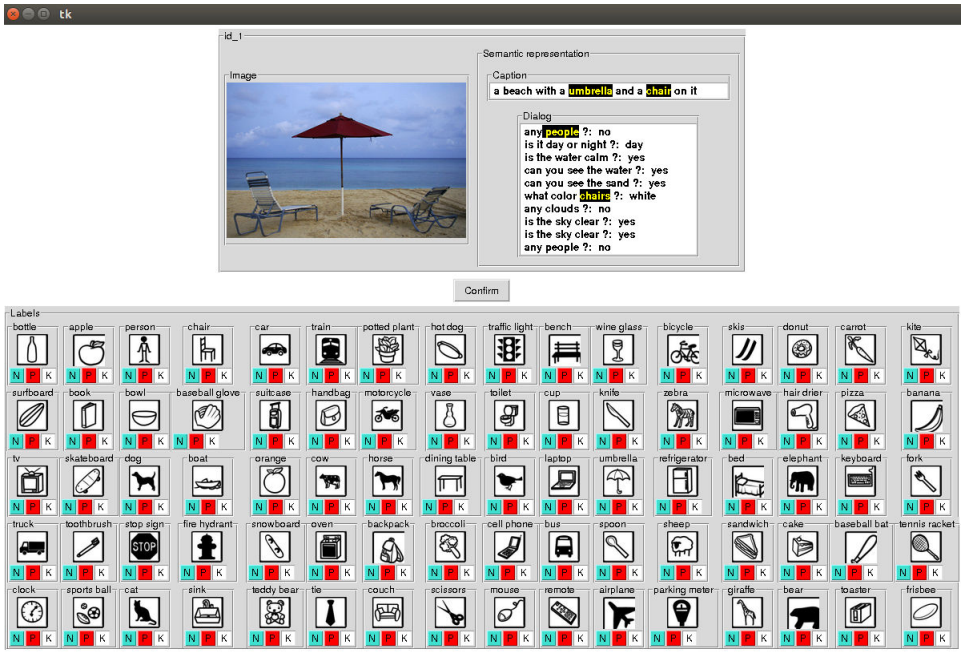


FIGURE 3.20 – Interface utilisateur pour collecter les annotations de prédiction d'échec. L'image est présentée avec sa représentation sémantique générée (description + dialogue). Chaque étiquette peut être annotée avec l'un des 3 états : faux négatif, faux positif et correct.

TABLE 3.5 – Statistiques de prédiction de défaillance.

	faux négatif		*faux positif*	
	#vrai	#prédit	#vrai	#prédit
VT.	614	-	588	-
utilisateurs	308	379	213	485
classifieur	250	490	180	530

MS-COCO 1 parmi 3 cas : i) *faux négatif*. La représentation sémantique a manqué l'étiquette (ex : description et dialogue ne mentionnent pas le cheval dans l'image). ii) *faux positif*. La représentation sémantique hallucine l'objet (par exemple, voir une voiture dans une scène de cuisine). iii) *correct*. L'algorithme a réussi à prédire l'étiquette, soit son absence, soit sa présence. Le tableau 3.5 montre le nombre de défaillances de la classification multi-étiquettes (614 faux négatifs et 588 faux positifs). Les sujets humains ont pu identifier la moitié des échecs (308/614 FN et 213/588 PF)

TABLE 3.6 – Classification multi-labels.

	label		*image*	
	mAP	%	mAP	%
pas de sélection	54.3	100	54.3	100
utilisateurs	84.2	96	86.1	53
classifieur	79.8	93	81.7	49
seuillage score	66.1	93	73.5	49

avec une précision de 60% environ (308/379 et 213/485).

Nous avons également développé un algorithme automatique de prédiction de défaillance, fournissant une capacité d'auto-diagnostic. Nous définissons un classifieur linéaire ternaire indépendant pour chaque classe avec 3 sorties possibles : $correct$, FN, FP. L'entrée est l'image I concaténée avec le dernier état caché de l'encodeur de représentation sémantique $[\{Q_k, A_k\}_{k=1:K}, c(I)]$. La vérité terrain est construite en comparant les résultats de la classification multi-labels et aux vraies classes. Le modèle est optimisé en utilisant un critère d'entropie croisée. l'algorithme automatique est moins précis (détecte $\approx 41\%$ de faux positifs avec $\approx 51\%$ de précision), mais a l'avantage de réduire l'effort humain.

La détection de défaillance peut être définie à l'aide de deux types d'expériences : *Rejet par label* : les étiquettes suspectes sont rejetées, les autres sont conservées. *Rejet d'image* : en cas d'étiquette suspecte, l'image est rejetée. La table 3.6 montre les deux expériences, et se lit comme suit : la performance de classification est de 54.3% lors de l'évaluation sur 100% de l'ensemble de test. Lorsque l'utilisateur rejette 4% des étiquettes, la performance atteint 84,2%. Lorsque notre algorithme de rejet conserve 93% des données, la précision est de 79,8%, ce qui est proche de la performance humaine. Nous constatons une forte amélioration de nos deux méthodes, confirmant la capacité de prédiction de défaillance du goulot d'étranglement sémantique. Les prédictions d'échec améliorent la

précision moyenne de 30% avec 4% d'images supprimées en moyenne pour chaque classe. Rejeter complètement l'image lorsqu'une seule mauvaise prédiction est suspectée permet d'améliorer les performances, mais avec une augmentation du nombre d'images rejetées. Nous montrons également dans la dernière ligne du tableau 3.6 la performance d'un algorithme de rejet élémentaire. Il est basé sur le seuillage du score de confiance produit par le classifieur multi-étiquettes, et ajusté pour atteindre le même taux de rejet que l'algorithme de détection de défaillance. Cet algorithme de seuillage permet une augmentation de performance plus faible comparée aux deux autres approches.

3.6 Conclusion

Avec l'avènement des techniques d'apprentissage profond en vision par ordinateur, la nécessité de compréhension de la décision de ces algorithmes est devenue indispensable.

Dans ce chapitre, nous avons introduit une nouvelle méthode pour représenter une image avec de l'information sémantique exprimée en langage naturel. Notre principale motivation était d'introduire un goulot d'étranglement intelligible dans le processus de traitement. Nous avons montré qu'en combinant et en adaptant plusieurs techniques de l'état de l'art, notre approche était capable de générer des descriptions textuelles riches qui peuvent se substituer aux images pour deux tâches de vision : la recherche d'images basée sur le contenu sémantique et la classification multi-étiquettes. La génération de la représentation se fait en premier lieu à l'aide d'un générateur de description. Puis des modules de sélection de questions et de réponses qui permet d'affiner l'information sémantique extraite de l'image.

Les résultats obtenus montrent que notre approche est capable de générer des représentations sémantiques qui donnent des résultats à l'état de l'art en matière de recherche d'images basée sur le contenu et fonctionnent également très bien sur la tâche de classification d'images.

Nous avons évalué quantitativement et qualitativement l'utilisation de ce goulot d'étranglement sémantique comme un outil de diagnostic pour détecter les défaillances dans le processus de prédiction. L'utilisateur du système peut examiner sur quelle base l'inférence a été réalisée et ainsi d'accepter ou de rejeter la décision suivant sa connaissance et son expérience humaine. Selon nous, cela contribue à une métrique plus claire de l'explicabilité, une préoccupation majeure en l'intelligence artificielle.

Conclusions et Perspectives

À travers ce manuscrit nous avons traité deux problématiques liées à la mise en pratique des algorithmes d'apprentissage automatique.

Pour se substituer en données visuelles de références, nous avons présenté deux approches de classification zero-shot. Dans le chapitre 1 nous utilisons l'apprentissage de métrique pour transformer l'espace sémantique initial en un espace optimal pour une tâche de classification visuelle. En réponse aux limitations des approches par plongements sémantique nous proposons dans le chapitre 2 de générer des caractéristiques visuelles pour les classes non vues.

Notre deuxième problématique concernait l'intelligibilité de la prise de décision d'un modèle statistique. Nous avons développé dans le chapitre 3, un goulot d'étranglement sémantique autorisant l'inspection des étapes intermédiaires de décision.

Nous présentons maintenant un résumé de nos contributions et de leurs résultats et concluons la thèse avec des perspectives pour des travaux futurs éventuels.

Résumé des contributions

Apprentissage d'un espace sémantique optimal

La question clé de ce chapitre est de trouver comment transformer l'ensemble des attributs initiaux en un espace optimal pour une tâche de classification zero-shot. Nous avons proposé d'améliorer cet ensemble à l'aide d'une métrique de Mahalanobis. La métrique est capable à la fois de sélectionner et de transformer la distribution des données d'origines. Elle est obtenue suivant un critère de coût multi-objectifs qui contraint en premier lieu le bon plongement sémantique de l'image et qui ensuite le transforme. Nous avons validé empiriquement l'idée que l'optimisation conjointe de ces deux critères permet de meilleures performances, en comparaison aux approches par plongement sémantique, sur deux tâches visuelles zero-shot : la reconnaissance et la recherche d'objet dans une base de données.

Génération de caractéristiques

Pour pallier les limitations des approches par plongement sémantique, nous avons proposé de générer des caractéristiques visuelles pour l'ensemble des classes non vues. Nous avons défini un générateur conditionné par la représentation sémantique d'une classe. Après optimisation notre approche permet de générer des exemples d'entraînement artificiels pour les catégories sans exemples. Les données vraies et artificielles sont utilisées ensuite pour l'apprentissage d'un classifieur discriminant. À travers un ensemble d'expériences nous avons montré que l'utilisation d'un classifieur discriminant appris avec les exemples générés permet la suppression du «hubness problem» et autorise la classification d'image pour la tâche de zero-shot généralisée.

Goulot d'étranglement sémantique

Dans ce chapitre nous avons abordé la question de l'intelligibilité des calculs pour les tâches de vision par ordinateur. Nous avons introduit une nouvelle méthode de représentation de l'image qui est par nature sémantique. Notre proposition est un goulot étranglement sémantique dans le processus de traitement, la représentation de l'image est exprimée entièrement en langage naturel.

Nous avons montré que notre approche est capable de générer des représentations sémantiques qui donnent des résultats au niveau de l'état de l'art pour une tâche de recherche d'images et qui fonctionnent également très bien pour la tâche de classification multi-étiquettes. Le point important de notre approche est la capacité de l'utilisateur à détecter des défaillances de prédiction à partir de la représentation. Cette détection manuelle permet d'accepter ou de rejeter une décision et ainsi d'améliorer les performances globales du système.

Perspectives

Dans les deux sections suivantes, nous proposons des perpectives possibles aux travaux présentés dans les chapitres 1, 2 et 3.

Reconnaissance visuelle sans exemple de référence Les deux chapitres consacrés à la classification zero-shot ont permis de répondre à notre problématique initiale de reconnaissance visuelle sans exemple de référence.

Le modèle proposé dans le chapitre 1 pourrait bénéficier d'une architecture plus complexe, notamment pour la détection des attributs, qui est indispensable à la bonne application de la métrique. Une autre amélioration mineure serait d'introduire une formulation par triplet [157] pour la fonction de coût. La création de l'ensemble des exemples par triplets est plus naturelle et pourrait permettre d'obtenir de meilleures performances de classification. Enfin comme mentionné dans le chapitre 2, l'espace des caractéristiques visuelles est plus adapté à une tâche de classification visuelle. Une évolution simple serait de réaliser le plongement inverse, et projeter les labels dans l'espace des caractéristiques visuelles. Dans la littérature récente les modèles adversaires ont connu un véritable engouement, une évolution directe du chapitre 2 serait de bénéficier de ces améliorations pour le développement d'un nouveau modèle génératif.

Une perspective intéressante en classification zero-shot est la nature de l'information sémantique utilisée pour représenter les classes d'objets. Les attributs peuvent être perçus comme trop rigides et donc nuire aux capacités de discrimination du modèle, notamment pour les classes proches visuellement et sémantiquement. Quant aux descriptions textuelles, elles permettent d'obtenir de meilleures performances de reconnaissance, mais demandent une supervision trop importante pour un passage à l'échelle. Comme constaté lors des expériences de classification à grande échelle, les représentations «word2vec» offrent un moyen de représenter les classes

sans effort de supervision, cependant les performances restent limitées. En effet aucune information visuelle explicite n'est fournie lors de l'apprentissage, la fonction de coût prenant uniquement en compte le contexte textuel. Pour une mise en pratique efficace, une solution serait de combiner les approches zero-shot avec un faible nombre d'exemples de référence. Comme constaté dans la section 1.4.4 les performances de reconnaissances visuelles augmentent significativement lorsque qu'un nombre réduit de données est introduit.

Intelligibilité de la décision Nous avons introduit une nouvelle méthode pour représenter des images avec un goulot d'étranglement intelligible dans le pipeline de traitement. Ce goulot d'étranglement permet à un utilisateur d'examiner sur quelle base l'inférence a été réalisée et ainsi d'accepter ou de rejeter la décision. Par exemple lors d'interrogations plus poussées, un médecin peut vouloir contrôler les caractéristiques de prédiction utilisées lors d'une analyse.

Une perspective intéressante serait de combiner de nouveaux types d'extracteurs, comme une visualisation par carte de saillance ou encore une extraction d'attributs sémantiques. Le mécanisme de fusion permettant l'ajout d'informations sémantiques supplémentaires peut se faire sans changement important dans le processus de décision.

Une dernière piste d'amélioration serait d'utiliser un mécanisme attentionnel sur la représentation textuelle. Ceci permettrait de fournir une indication sur l'importance de chaque mot lors de la prise de décision.

Conférences internationales avec comité de revue

- **Semantic bottleneck for computer vision tasks**
  Maxime Bucher, Stéphane Herbin, Frédéric Jurie
  Asian Conference on Computer Vision (ACCV) 2018
- [22] **Generating visual representations for zero-shot classification**
  Maxime Bucher, Stéphane Herbin, Frédéric Jurie
  International Conference on Computer Vision (ICCV) Workshops 2017 **(Prix du meilleur article)**
- [24] **Improving semantic embedding consistency by metric learning for zero-shot classiffication**
  Maxime Bucher, Stéphane Herbin, Frédéric Jurie
  European Conference on Computer Vision (ECCV) 2016
- [25] **Hard negative mining for metric learning based zero-shot classification**
  Maxime Bucher, Stéphane Herbin, Frédéric Jurie
  European Conference on Computer Vision (ECCV) Workshops 2016

Conférences nationales avec comité de revue

- [23] **Zero-Shot Classification by Generating Artificial Visual Features**
  Maxime Bucher, Stéphane Herbin, Frédéric Jurie
  Reconnaissance des Formes, Image, Apprentissage et Perception (RFIAP) 2018
- **Apprentissage d'un espace sémantique optimal pour la reconnaissance d'objets sans exemple d'apprentissage**
  Maxime Bucher, Stéphane Herbin, Frédéric Jurie
  Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS) 2017

Bibliographie

[1] *Apprentissage avec un nombre réduit d'exemple*.
https://medium.com/sap-machine-learning-research/deep-few-shot-learning-a1caa289f18.

[2] *Image de segway*.
http://www.segway.com/media/2253/homepageproduct_i2secommercial_v3.png.

[3] *Image des catégories possibles*.
https://i.vimeocdn.com/video/567983513_1280x720.jpg.

[4] *Image des tâches de vision par ordinateur*.
http://web.stanford.edu/class/cs224n/.

[5] *Visualisation des couches intermédiaires*.
https://distill.pub/2017/feature-visualization/.

[6] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, *Label-Embedding for Image Classification*, IEEE Trans. on Pattern Analysis and Machine Intelligence, (2015).

[7] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, *Evaluation of Output Embeddings for Fine-Grained Image Classification*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[8] H. ALI, Y. CHALI, AND S. A. HASAN, *Automation of question generation from sentences*, in The Third Workshop on Question Generation, 2010.

[9] S. ANTOL, A. AGRAWAL, J. LU, M. MITCHELL, D. BATRA, C. L. ZITNICK, AND D. PARIKH, *VQA : Visual Question Answering*, in ICCV, 2015.

[10] M. ARJOVSKY, S. CHINTALA, AND L. BOTTOU, *Wasserstein gan*, International Conference on Machine Learning, (2017).

[11] G. ARORA, V. K. VERMA, A. MISHRA, AND P. RAI, *Generalized zero-shot learning via synthesized examples*, Proceedings of the IEEE conference on computer vision and pattern recognition, (2018).

[12] H. ARORA, N. LOEFF, D. A. FORSYTH, AND N. AHUJA, *Unsupervised segmentation of objects using efficient learning*, in Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, 2007, pp. 1–7.

[13] N. AUDEBERT, B. LE SAUX, AND S. LEFÈVRE, *Semantic segmentation of earth observation data using multimodal and multi-scale deep networks*, in Asian Conference on Computer Vision, 2016.

[14] L. J. BA, K. SWERSKY, S. FIDLER, AND R. SALAKHUTDINOV, *Predicting deep zero-shot convolutional neural networks using textual descriptions*, in 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, 2015, pp. 4247–4255.

[15] M. B. BADER-EL-DEN, E. TEITEI, AND M. ADDA, *Hierarchical classification for dealing with the Class imbalance problem.*, IJCNN, (2016).

[16] D. BAU, B. ZHOU, A. KHOSLA, A. OLIVA, AND A. TORRALBA, *Network dissection : Quantifying interpretability of deep visual representations*, in CVPR, 2017.

[17] H. BEN-YOUNES, R. CADENE, M. CORD, AND N. THOME, *Mutan : Multimodal tucker fusion for visual question answering*, in Proceedings of the IEEE international conference on computer vision, 2017.

[18] Y. BENGIO, L. YAO, G. ALAIN, AND P. VINCENT, *Generalized denoising auto-encoders as generative models*, in Advances in Neural Information Processing Systems, 2013, pp. 899–907.

[19] T. L. BERG, A. C. BERG, AND J. SHIH, *Automatic attribute discovery and characterization from noisy web data*, in European Conference on Computer Vision (ECCV), 2010.

[20] B. BHATTARAI, G. SHARMA, AND F. JURIE, *Cp-mtml : Coupled projection multi-task metric learning for large scale face retrieval*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[21] O. BIRAN AND C. COTTON, *Explanation and justification in ml : A survey*, in IJCAI, 2017.

[22] M. BUCHER, S. HERBIN, AND F. JURIE, *Generating visual representations for zero-shot classification*, in International Conference on Computer Vision, 2017.

[23] M. BUCHER, S. HERBIN, AND F. JURIE, *Zero-Shot Classification by Generating Artificial Visual Features*, in RFIAP, 2018.

[24] M. BUCHER, S. HERBIN , AND F. JURIE, *Improving semantic embedding consistency by metric learning for zero-shot classiffication*, in ECCV, 2016.

[25] V. M. BUCHER, S. HERBIN, AND F. JURIE, *Hard negative mining for metric learning based zero-shot classification*, in Computer Vision–ECCV 2016 Workshops, Springer, 2016, pp. 524–531.

[26] M. E. CABRERA AND J. P. WACHS, *Embodied gesture learning from one-shot*, in 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN, IEEE, 2016, pp. 1092–1097.

[27] M. CAMPO, J. ESPINOZA, J. RIEGER, AND A. TALIYAN, *Collaborative metric learning recommendation system : Application to theatrical movie releases*, KDD, (2018).

[28] O. CANÉVET AND F. FLEURET, *Efficient sample mining for object detection.*, in ACML, 2014.

[29] S. CHANGPINYO, W.-L. CHAO, B. GONG, AND F. SHA, *Synthesized classifiers for zero-shot learning*, in Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on, IEEE, 2016, pp. 5327–5336.

[30] W.-L. CHAO, S. CHANGPINYO, B. GONG, AND F. SHA, *An empirical study and analysis of generalized zero-shot learning for object recognition in the wild*, in European Conference on Computer Vision, Springer, 2016, pp. 52–68.

[31] O. CHAPELLE, P. HAFFNER, AND V. N. VAPNIK, *Support vector machines for histogram-based image classification*, transactions on Neural Networks, (1999).

[32] W. CHEN, *Aist, g., mostow, j. : Generating questions automatically from informational text*, in AIED, 2009.

[33] R. G. CINBIS, J. VERBEEK, AND C. SCHMID, *Unsupervised metric learning for face identification in tv video*, in Computer Vision (ICCV), 2011 IEEE International Conference on, 2011.

[34] G. CSURKA, *Domain adaptation for visual applications : A comprehensive survey*, Advances in Computer Vision and Pattern Recognition, (2017).

[35] B. DAI AND D. LIN, *Contrastive Learning for Image Captioning*, in NIPS, 2017.

[36] B. DAI, D. LIN, R. URTASUN, AND S. FIDLER, *Towards Diverse and Natural Image Descriptions via a Conditional GAN.*, in CVPR, 2017.

[37] A. DAS, S. KOTTUR, K. GUPTA, A. SINGH, D. YADAV, J. M. F. MOURA, D. PARIKH, AND D. BATRA, *Visual dialog*, in CVPR, 2016.

[38] A. DAS, S. KOTTUR, J. M. F. MOURA, S. LEE, AND D. BATRA, *Learning cooperative visual dialog agents with deep reinforcement learning*, in ICCV 2017, 2017.

[39] J. DENG, W. DONG, R. SOCHER, L.-J. LI, K. LI, AND L. FEI-FEI, *Imagenet : A large-scale hierarchical image database*, in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 248–255.

[40] D. DORAN, S. SCHULZ, AND T. R. BESOLD, *What does explainable ai really mean ? a new conceptualization of perspectives*, Comprehensibility and Explanation in AI and ML (CEX), (2017).

[41] F. DOSHI-VELEZ AND B. KIM, *A roadmap for a rigorous science of interpretability*, arXiv :1702.08608, (2017).

[42] A. DOSOVITSKIY AND T. BROX, *Inverting visual representations with convolutional networks*, in CVPR, 2016.

[43] M. DOUZE, A. SZLAM, B. HARIHARAN, AND H. JÉGOU, *Low-shot learning with large-scale diffusion*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2018).

[44] K. DUAN, D. PARIKH, D. CRANDALL, AND K. GRAUMAN, *Discovering localized attributes for fine-grained recognition*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

[45] T. DURAND, T. MORDAN, N. THOME, AND M. CORD, *Wildcat : Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[46] C. EGGERT, A. WINSCHEL, AND R. LIENHART, *On the Benefit of Synthetic Data for Company Logo Detection.*, in ACM Multimedia, 2015.

[47] M. ELHOSEINY, B. SALEH, AND A. ELGAMMAL, *Write a Classifier : Zero-Shot Learning Using Purely Textual Descriptions*, in IEEE International Conference on Computer Vision (ICCV), 2013.

[48] M. EVERINGHAM, L. VAN GOOL, C. K. I. WILLIAMS, J. WINN, AND A. ZISSERMAN, *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*, 2012.

[49] N. FARAJI DAVAR, T. DE CAMPOS, D. WINDRIDGE, J. KITTLER, AND W. CHRISTMAS, *Domain adaptation in the context of sport video action recognition*, in Domain Adaptation Workshop, in conjunction with NIPS, 2011.

[50] A. FARHADI, I. ENDRES, D. HOIEM, AND D. FORSYTH, *Describing objects by their attributes*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[51] A. FROME, G. S. CORRADO, J. SHLENS, S. BENGIO, J. DEAN, M. RANZATO, AND T. MIKOLOV, *DeViSE : A Deep Visual-Semantic Embedding Model.*, in Conference on Neural Information Processing Systems (NIPS), 2013.

[52] Y. FU, T. M. HOSPEDALES, T. XIANG, Z. FU, AND S. GONG, *Transductive multi-view embedding for zero-shot recognition and annotation*, in European Conference on Computer Vision (ECCV), 2014.

[53] Y. FU, X. ZHU, AND B. LI, *A survey on instance selection for active learning*, Knowledge and Information Systems, 35 (2013), pp. 249–283.

[54] Z. FU, T. A. XIANG, E. KODIROV, AND S. GONG, *Zero-shot object recognition by semantic manifold distance*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[55] S. GANJU, O. RUSSAKOVSKY, AND A. GUPTA, *What's in a question : Using visual questions as a form of supervision*, in CVPR, 2017.

[56] L. H. GILPIN, D. BAU, B. Z. YUAN, A. BAJWA, M. SPECTER, AND L. KAGAL, *Explaining explanations : An approach to evaluating interpretability of ml*, arXiv :1806.00069, (2018).

[57] X. GLOROT, A. BORDES, AND Y. BENGIO, *Deep sparse rectifier neural networks*, in Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, 2011, pp. 315–323.

[58] I. GOODFELLOW, J. POUGET-ABADIE, M. MIRZA, B. XU, D. WARDE-FARLEY, S. OZAIR, A. COURVILLE, AND Y. BENGIO, *Generative adversarial nets*, in Advances in neural information processing systems, 2014, pp. 2672–2680.

[59] A. GORDO AND D. LARLUS, *Beyond instance-level image retrieval : Leveraging captions to learn a global visual representation for semantic retrieval*, in CVPR, 2017.

[60] A. GRETTON, D. SEJDINOVIC, H. STRATHMANN, S. BALAKRISHNAN, M. PONTIL, K. FUKUMIZU, AND B. K. SRIPERUMBUDUR, *Optimal kernel choice for large-scale two-sample tests*, in Advances in neural information processing systems, 2012, pp. 1205–1213.

[61] R. GUIDOTTI, A. MONREALE, F. TURINI, D. PEDRESCHI, AND F. GIANNOTTI, *A survey of methods for explaining black box models*, ACM Computing Surveys, (2018).

[62] H. GUO AND H. L. VIKTOR, *Learning from imbalanced data sets with boosting and data generation - the DataBoost-IM approach.*, SIGKDD Explorations, (2004).

[63] J. HAMM AND M. BELKIN, *Probabilistic Zero-shot Classification with Semantic Rankings*, The Conference on Uncertainty in Artificial Intelligence (UAI), (2017).

[64] J. HE, M. LI, H.-J. ZHANG, H. TONG, AND C. ZHANG, *Manifold-ranking based image retrieval*, in Proceedings of the 12th annual ACM international conference on Multimedia, 2004.

[65] K. HE, X. ZHANG, S. REN, AND J. SUN, *Deep residual learning for image recognition*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[66] X. He, O. King, W.-Y. Ma, M. Li, and H.-J. Zhang, *Learning a semantic space from user's relevance feedback for image retrieval*, IEEE transactions on Circuits and Systems for Video technology, (2003).

[67] X. He, W.-Y. Ma, and H.-J. Zhang, *Learning an image manifold for retrieval*, in Proceedings of the 12th annual ACM international conference on Multimedia, 2004.

[68] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell, *Generating visual explanations*, in ECCV, 2016.

[69] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, Neural computation, (1997).

[70] F. M. Hohman, M. Kahng, R. Pienta, and D. H. Chau, *Visual analytics in deep learning : An interrogative survey for the next frontiers*, TVCG, (2018).

[71] S. C. Hoi, W. Liu, and S.-F. Chang, *Semi-supervised distance metric learning for collaborative image retrieval and clustering*, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), (2010).

[72] A. Jabri, A. Joulin, and L. van der Maaten, *Revisiting vqa baselines*, in ECCV, 2016.

[73] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, *Reading Text in the Wild with Convolutional Neural Networks*, International Journal of Computer Vision, 116 (2016), pp. 1–20.

[74] D. Jayaraman and K. Grauman, *Zero-shot recognition with unreliable attributes*, in Conference on Neural Information Processing Systems (NIPS), 2014.

[75] A. JOULIN, L. VAN DER MAATEN, A. JABRI, AND N. VASILACHE, *Learning visual features from large weakly supervised data*, in European Conference on Computer Vision, Springer, 2016, pp. 67–84.

[76] B. KAYALIBAY, G. JENSEN, AND P. VAN DER SMAGT, *Cnn-based segmentation of medical imaging data*, arXiv preprint arXiv :1701.03056, (2017).

[77] P.-J. KINDERMANS, S. HOOKER, J. ADEBAYO, M. ALBER, K. T. SCHÜTT, S. DÄHNE, D. ERHAN, AND B. KIM, *The (un) reliability of saliency methods*, NIPS, (2017).

[78] D. P. KINGMA AND J. BA, *Adam : A method for stochastic optimization*, International Conference on Learning Representations (ICLR), (2014).

[79] E. KODIROV, T. XIANG, Z. FU, AND S. GONG, *Unsupervised Domain Adaptation for Zero-Shot Learning*, in IEEE International Conference on Computer Vision (ICCV), 2015.

[80] E. KODIROV, T. XIANG, AND S. GONG, *Semantic autoencoder for zero-shot learning*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3174–3183.

[81] T. KOOI, G. LITJENS, B. VAN GINNEKEN, A. GUBERN-MÉRIDA, C. I. SÁNCHEZ, R. MANN, A. DEN HEETEN, AND N. KARSSEMEIJER, *Large scale deep learning for computer aided detection of mammographic lesions*, Medical image analysis, (2017).

[82] J. KRAPAC, J. VERBEEK, AND F. JURIE, *Modeling spatial layout with fisher vectors for image categorization*, in Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, 2011.

[83] A. KRIZHEVSKY, I. SUTSKEVER, AND G. E. HINTON, *ImageNet Classification with Deep Convolutional Neural Networks.*, in Conference on Neural Information Processing Systems (NIPS), 2012, pp. 1106–1114.

[84] C. H. LAMPERT, H. NICKISCH, AND S. HARMELING, *Attribute-Based Classification for Zero-Shot Visual Object Categorization.*, IEEE Trans. on Pattern Analysis and Machine Intelligence, 36 (2014), pp. 453–465.

[85] C. LE BARZ, N. THOME, M. CORD, S. HERBIN, AND M. SANFOURCHE, *Absolute geo-localization thanks to hidden markov model and exemplar-based metric learning*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015.

[86] C. V. LE BARZ, N. THOME, M. CORD, S. HERBIN, AND M. SANFOURCHE, *Exemplar based metric learning for robust visual localization*, in IEEE International Conference on Image Processing, ICIP 2015, 2015, pp. 388–w6mP.

[87] Y. LECUN, L. BOTTOU, Y. BENGIO, AND P. HAFFNER, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE, 86 (1998), pp. 2278–2324.

[88] X. LI, Y. GUO, AND D. SCHUURMANS, *Semi-Supervised Zero-Shot Classification With Label Representation Learning*, in IEEE International Conference on Computer Vision (ICCV), 2015.

[89] X. LI, C. M. SNOEK, M. WORRING, D. KOELMA, AND A. W. SMEULDERS, *Bootstrapping visual categorization with relevant negatives*, IEEE Transactions on Multimedia, 15 (2013), pp. 933–945.

[90] Y. LI, C. HUANG, X. TANG, AND C. CHANGE LOY, *Learning to disambiguate by asking discriminative questions*, in CVPR, 2017.

[91] Y. LI, K. SWERSKY, AND R. ZEMEL, *Generative moment matching networks*, in International Conference on Machine Learning, 2015, pp. 1718–1727.

[92] T.-Y. LIN, M. MAIRE, S. BELONGIE, J. HAYS, P. PERONA, D. RAMANAN, P. DOLLÁR, AND C. L. ZITNICK, *Microsoft coco : Common objects in context*, in ECCV, 2014.

[93] X. LIN AND D. PARIKH, *Leveraging visual question answering for image-caption ranking*, in ECCV, 2016.

[94] Z. C. LIPTON, *The mythos of model interpretability*, in ICML, 2016.

[95] M. LONG, J. WANG, G. DING, J. SUN, AND P. S. YU, *Transfer feature learning with joint distribution adaptation*, in Proceedings of the IEEE international conference on computer vision, 2013, pp. 2200–2207.

[96] J. LU, A. KANNAN, J. YANG, D. PARIKH, AND D. BATRA, *Best of both worlds : Transferring knowledge from discriminative learning to a generative visual dialog model*, NIPS, (2017).

[97] J. LU, C. XIONG, D. PARIKH, AND R. SOCHER, *Knowing When to Look : Adaptive Attention via a Visual Sentinel for Image Captioning*, in CVPR, 2017.

[98] A. L. MAAS, A. Y. HANNUN, AND A. Y. NG, *Rectifier nonlinearities improve neural network acoustic models*, in ICML, 2013.

[99] L. V. D. MAATEN AND G. HINTON, *Visualizing data using t-sne*, Journal of machine learning research, 9 (2008), pp. 2579–2605.

[100] D. K. MAHAJAN, S. SELLAMANICKAM, AND V. NAIR, *A joint learning framework for attribute models and object descriptions.*, in IEEE International Conference on Computer Vision (ICCV), 2011.

[101] A. MAHENDRAN AND A. VEDALDI, *Understanding deep image representations by inverting them*, in CVPR, 2015.

[102] A. MAKHZANI, J. SHLENS, N. JAITLY, I. GOODFELLOW, AND B. FREY, *Adversarial autoencoders*, ICLR, (2016).

[103] M. MALINOWSKI, M. ROHRBACH, AND M. FRITZ, *Ask your neurons : A neural-based approach to answering questions about images*, in ICCV, 2015.

[104] B. MCFEE AND G. R. LANCKRIET, *Metric learning to rank*, in Proceedings of the 27th International Conference on Machine Learning (ICML-10), 2010, pp. 775–782.

[105] A. MEHROTRA AND A. DUKKIPATI, *Generative adversarial residual pairwise networks for one shot learning*, arXiv preprint arXiv :1703.08033, (2017).

[106] P. MELVILLE AND R. J. MOONEY, *Constructing Diverse Classifier Ensembles using Artificial Training Examples.*, IJCAI, (2003).

[107] T. MENSINK, E. GAVVES, AND C. G. M. SNOEK, *COSTA : Co-Occurrence Statistics for Zero-Shot Classification.*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[108] T. MENSINK, J. VERBEEK, F. PERRONNIN, AND G. CSURKA, *Metric learning for large scale image classification : Generalizing to new classes at near-zero cost*, in Computer Vision–ECCV 2012, Springer, 2012, pp. 488–501.

[109] T. MIKOLOV, K. CHEN, G. CORRADO, AND J. DEAN, *Efficient estimation of word representations in vector space*, In ICLR, (2013).

[110] T. MIKOLOV, I. SUTSKEVER, K. CHEN, G. S. CORRADO, AND J. DEAN, *Distributed representations of words and phrases and their compositionality*, in Advances in neural information processing systems, 2013, pp. 3111–3119.

[111] M. MIRZA AND S. OSINDERO, *Conditional generative adversarial nets*, NIPS, (2014).

[112] G. MONTAVON, W. SAMEK, AND K. R. MÜLLER, *Methods for interpreting and understanding deep neural networks*, Digital Signal Processing : A Review Journal, (2018).

[113] T. MORDAN, N. THOME, G. HENAFF, AND M. CORD, *End-to-End Learning of Latent Deformable Part-Based Representations for Object Detection*, International Journal of Computer Vision, (2018).

[114] N. MOSTAFAZADEH, I. MISRA, J. DEVLIN, M. MITCHELL, X. HE, AND L. VANDERWENDE, *Generating natural questions about an image*, ACL, (2016).

[115] R. NEGREL, A. LECHERVY, AND F. JURIE, *MLBoost Revisited : A Faster Metric Learning Algorithm for Identity-Based Face Retrieval*, in BMVC : British Machine Vision Conference 2016, 2016.

[116] M.-E. NILSBACK AND A. ZISSERMAN, *Automated flower classification over a large number of classes*, in Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on, IEEE, 2008, pp. 722–729.

[117] M. NOROUZI, T. MIKOLOV, S. BENGIO, Y. SINGER, J. SHLENS, A. FROME, G. S. CORRADO, AND J. DEAN, *Zero-Shot Learning by Convex Combination of Semantic Embeddings*, in International Conference on Learning Representations (ICLR), Dec. 2013.

[118] J. OGIER DU TERRAIL AND F. JURIE, *ON THE USE OF DEEP NEURAL NETWORKS FOR THE DETECTION OF SMALL VEHICLES IN ORTHO-IMAGES*, in IEEE International Conference on Image Processing, 2017.

[119] C. OLAH, A. SATYANARAYAN, I. JOHNSON, S. CARTER, L. SCHUBERT, K. YE, AND A. MORDVINTSEV, *The building blocks of interpretability*, Distill, (2018).

[120] M. OQUAB, L. BOTTOU, I. LAPTEV, AND J. SIVIC, *Learning and transferring mid-level image representations using convolutional neural networks*, in Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 1717–1724.

[121] D. PARIKH AND K. GRAUMAN, *Interactively building a discriminative vocabulary of nameable attributes*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[122] D. PARIKH AND K. GRAUMAN, *Relative attributes*, in IEEE International Conference on Computer Vision (ICCV), 2011.

[123] D. H. PARK, L. A. HENDRICKS, Z. AKATA, A. ROHRBACH, B. SCHIELE, T. DARRELL, AND M. ROHRBACH, *Multimodal explanations : Justifying decisions and pointing to the evidence*, 2018.

[124] G. PATTERSON, C. XU, H. SU, AND J. HAYS, *The SUN Attribute Database : Beyond Categories for Deeper Scene Understanding*, International Journal of Computer Vision (IJCV), 108 (2014), pp. 59–81.

[125] J. PENNINGTON, R. SOCHER, AND C. MANNING, *Glove : Global vectors for word representation*, in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.

[126] F. PERRONNIN AND C. DANCE, *Fisher kernels on visual vocabularies for image categorization*, in CVPR, 2007.

[127] N. F. RAJANI AND R. J. MOONEY, *Using explanations to improve ensembling of visual question answering systems*, in IJCAI, 2017.

[128] G. RAS, P. HASELAGER, AND M. VAN GERVEN, *Explanation methods in deep learning : Users, values, concerns and challenges*, Explainable and Interpretable Models in Computer Vision and Machine Learning, (2018).

[129] S. RAVI AND H. LAROCHELLE, *Optimization as a model for few-shot learning*, ICLR, (2017).

[130] S. REED, Z. AKATA, X. YAN, L. LOGESWARAN, B. SCHIELE, AND H. LEE, *Generative adversarial text to image synthesis*, ICML, (2016).

[131] M. ROHRBACH, M. STARK, AND B. SCHIELE, *Evaluating knowledge transfer and zero-shot learning in a large-scale setting*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[132] B. ROMERA-PAREDES AND P. H. TORR, *An embarrassingly simple approach to zero-shot learning*, in Proceedings of the International Conference on Machine learning, 2015, pp. 2152–2161.

[133] O. RUSSAKOVSKY, J. DENG, H. SU, J. KRAUSE, S. SATHEESH, S. MA, Z. HUANG, A. KARPATHY, A. KHOSLA, M. BERNSTEIN, ET AL., *Imagenet large scale visual recognition challenge*, IJCV, (2015).

[134] K. SAENKO, B. KULIS, M. FRITZ, AND T. DARRELL, *Adapting visual category models to new domains*, in European conference on computer vision, Springer, 2010, pp. 213–226.

[135] W. SAMEK, A. BINDER, G. MONTAVON, S. LAPUSCHKIN, AND K.-R. MÜLLER, *Evaluating the visualization of what a deep neural network has learned*, transactions on neural networks and learning systems, (2017).

[136] R. R. SELVARAJU, M. COGSWELL, A. DAS, R. VEDANTAM, D. PARIKH, AND D. BATRA, *Grad-cam : Visual explanations from deep networks via gradient-based localization*, in ICCV, 2017.

[137] P. H. SEO, A. LEHRMANN, B. HAN, AND L. SIGAL, *Visual reference resolution using attention memory for visual dialog*, NIPS, (2017).

[138] I. V. SERBAN, A. GARCÍA-DURÁN, C. GULCEHRE, S. AHN, S. CHANDAR, A. COURVILLE, AND Y. BENGIO, *Generating factoid questions with recurrent neural networks : The 30m factoid question-answer corpus*, ACL, (2016).

[139] S. SHALEV-SHWARTZ, Y. SINGER, AND A. Y. NG, *Online and batch learning of pseudo-metrics*, in Proceedings of the International Conference on Machine learning, ACM, 2004, p. 94.

[140] S. SHEKHAR, V. M. PATEL, H. V. NGUYEN, AND R. CHELLAPPA, *Generalized domain-adaptive dictionaries*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 361–368.

[141] Y. SHIGETO, I. SUZUKI, K. HARA, M. SHIMBO, AND Y. MATSUMOTO, *Ridge regression, hubness, and zero-shot learning*, in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2015, pp. 135–151.

[142] A. SHRIVASTAVA, A. GUPTA, AND R. GIRSHICK, *Training region-based object detectors with online hard example mining*, CVPR, (2016).

[143] K. SIMONYAN AND A. ZISSERMAN, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, in ICLR, 2014.

[144] J. SIVIC AND A. ZISSERMAN, *Video Google : A text retrieval approach to object matching in videos*, in ICCV, 2003.

[145] R. SOCHER, M. GANJOO, C. D. MANNING, AND A. NG, *Zero-Shot Learning Through Cross-Modal Transfer*, in Conference on Neural Information Processing Systems (NIPS), 2013.

[146] N. SRIVASTAVA, G. HINTON, A. KRIZHEVSKY, I. SUTSKEVER, AND R. SALAKHUTDINOV, *Dropout : A simple way to prevent neural networks from overfitting*, The Journal of Machine Learning Research, 15 (2014), pp. 1929–1958.

[147] S. SUKHBAATAR, J. BRUNA, M. PALURI, L. BOURDEV, AND R. FERGUS, *Training convolutional networks with noisy labels*, ICLR, (2015).

[148] C. SZEGEDY, W. LIU, Y. JIA, P. SERMANET, S. REED, D. ANGUELOV, D. ERHAN, V. VANHOUCKE, AND A. RABINOVICH, *Going*

deeper with convolutions, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[149] Y. TAMAAZOUSTI, H. L. BORGNE, C. HUDELOT, M. E. A. SEDDIK, AND M. TAMAAZOUSTI, *Learning more universal representations for transfer-learning*, Journal of IEEE Transactions on Pattern Analysis and Machine Intelligence, (2018).

[150] N. VERMA, D. MAHAJAN, S. SELLAMANICKAM, AND V. NAIR, *Learning hierarchical similarity metrics*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

[151] A. VEZHNEVETS, V. FERRARI, AND J. M. BUHMANN, *Weakly supervised semantic segmentation with a multi-image model*, CVPR, (2011).

[152] O. VINYALS, A. TOSHEV, S. BENGIO, AND D. ERHAN, *Show and tell : Lessons learned from the 2015 mscoco image captioning challenge*, TPAMI, (2017).

[153] W. VORAVUTHIKUNCHAI, B. CRÉMILLEUX, AND F. JURIE, *Histograms of pattern sets for image classification and object recognition*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014.

[154] C. WAH, S. BRANSON, P. WELINDER, P. PERONA, AND S. BELONGIE, *The Caltech-UCSD Birds-200-2011 Dataset*, tech. rep., July 2011.

[155] Q. WANG AND K. CHEN, *Zero-shot visual recognition via bidirectional latent embedding*, IJCV, (2017).

[156] Y. WANG AND G. MORI, *A Discriminative Latent Model of Object Classes and Attributes.*, in European Conference on Computer Vision (ECCV), 2010.

[157] K. Q. WEINBERGER AND L. K. SAUL, *Distance metric learning for large margin nearest neighbor classification*, Journal of Machine Learning Research, (2009).

[158] J. WESTON, S. BENGIO, AND N. USUNIER, *WSABIE : scaling up to large vocabulary image annotation*, in IJCAI, 2011, pp. 2764–2770.

[159] P. WU, S. C. HOI, P. ZHAO, C. MIAO, AND Z.-Y. LIU, *Online multi-modal distance metric learning with application to image retrieval*, in ieee transactions on knowledge and data engineering, 2016.

[160] Q. WU, P. WANG, C. SHEN, A. DICK, AND A. VAN DEN HENGEL, *Ask me anything : Free-form visual question answering based on knowledge from external sources*, in CVPR, 2016.

[161] S. WU, S. BONDUGULA, F. LUISIER, X. ZHUANG, AND P. NATARAJAN, *Zero-Shot Event Detection Using Multi-modal Fusion of Weakly Supervised Concepts.*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[162] Y. XIAN, Z. AKATA, G. SHARMA, Q. NGUYEN, M. HEIN, AND B. SCHIELE, *Latent embeddings for zero-shot classification*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 69–77.

[163] Y. XIAN, C. H. LAMPERT, B. SCHIELE, AND Z. AKATA, *Zero-shot learning-a comprehensive evaluation of the good, the bad and*

the ugly, IEEE Transactions on Pattern Analysis and Machine Intelligence, (2018).

[164] Y. XIAN, T. LORENZ, B. SCHIELE, AND Z. AKATA, *Feature generating networks for zero-shot learning*, Proceedings of the IEEE conference on computer vision and pattern recognition, (2018).

[165] M. YAMADA, L. SIGAL, AND M. RAPTIS, *No bias left behind : Covariate shift adaptation for discriminative 3d pose estimation*, in European Conference on Computer Vision, Springer, 2012, pp. 674–687.

[166] R. YAN, A. G. HAUPTMANN, AND R. JIN, *Negative pseudo-relevance feedback in content-based video retrieval*, in Proceedings of the eleventh ACM international conference on Multimedia, ACM, 2003, pp. 343–346.

[167] D. YOO, H. FAN, V. N. BODDETI, AND K. M. KITANI, *Efficient k-shot learning with regularized deep networks*, AAAI, (2018).

[168] F. X. YU, L. CAO, R. S. FERIS, J. R. SMITH, AND S.-F. F. CHANG, *Designing category-level attributes for discriminative visual recognition*, in IEEE International Conference on Computer Vision (ICCV), IEEE, 2013.

[169] M. D. ZEILER AND R. FERGUS, *Visualizing and understanding convolutional networks*, in ECCV, 2014.

[170] P. ZHANG, J. WANG, A. FARHADI, M. HEBERT, AND D. PARIKH, *Predicting failures of vision systems*, in CVPR, 2014.

[171] Q. ZHANG, R. CAO, F. SHI, Y. N. WU, AND S.-C. ZHU, *Interpreting cnn knowledge via an explanatory graph*, AAAI, (2018).

[172] Q. ZHANG, Y. YANG, Y. N. WU, AND S.-C. ZHU, *Interpreting cnns via decision trees*, arXiv :1802.00121, (2018).

[173] Z. ZHANG AND V. SALIGRAMA, *Zero-Shot Learning via Semantic Similarity Embedding.*, in IEEE International Conference on Computer Vision (ICCV), 2015.

[174] Z. ZHANG AND V. SALIGRAMA , *Zero-shot recognition via structured prediction*, in ECCV, 2016.

[175] Z. V. ZHANG AND V. SALIGRAMA, *Zero-shot learning via joint latent similarity embedding*, in IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 6034–6042.

[176] Y. ZHU, O. GROTH, M. BERNSTEIN, AND L. FEI-FEI, *Visual7w : Grounded question answering in images*, in CVPR, 2016.

[177] Y. ZHU, J. J. LIM, AND L. FEI-FEI, *Knowledge acquisition for visual question answering via iterative querying*, in CVPR, 2017.