Attention spatiale artificielle pour des modèles profonds interprétables de qualité embryonnaire

Tristan Gomez

DOCTORAT BRETAGNE LOIRE / MATHSTIC

Nantes Université

THÈSE DE DOCTORAT DE

NANTES UNIVERSITÉ

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *MathSTIC - Informatique*

Par
Tristan GOMEZ

Attention spatiale artificielle pour des modèles profonds interprétables de qualité embryonnaire

Thèse présentée et soutenue à Nantes Université, le 08/11/2022
Unité de recherche : Laboratoire des Sciences du Numérique de Nantes

Rapporteurs avant soutenance :

Camille Kurtz	Maître de conférences HDR - Université Paris cité
Clément Chatelain	Maître de conférences HDR - INSA Rouen Normandie

Composition du Jury :

Président :	Jean-Baptiste Fasquel	Professeur des universités - Université d'Angers
Examinatrice :	Jenny Benois-Pineau	Professeure des universités - Université de Bordeaux
Dir. de thèse :	Harold Mouchère	Professeur des universités - Nantes Université
	Thomas Fréour	Professeur des universités - CHU de Nantes

"An important feature of a learning machine is that its teacher will often be very largely ignorant of quite what is going on inside."

A.M. Turing, Computing Machinery and Intelligence, 1950

REMERCIEMENTS

Tout d'abord, je ne remercierai jamais assez Harold Mouchère et Thomas Fréour, mes directeurs de thèse, pour m'avoir fait confiance, m'avoir guidé, écouté et soutenu, ainsi que pour leurs conseils et leur savoir qu'ils ont partagé avec moi.

Je remercie Camille Kurtz et Clément Chatelain qui ont accepté de rapporter ma thèse, ainsi que l'ensemble du jury : Jenny Benois-Pineau et Jean-Baptiste Fasquel. Je tiens à remercier l'équipe IPI pour sa bonne humeur, ses pots et cet esprit de groupe particulier qu'elle apporte à ses membres. En particulier je remercie Gaëlle Jouis pour le soutien, les discussions passionnantes et plus généralement tout les moments de nos thèses respectives que nous avons partagé. Je voudrais aussi remercier les assistantes de l'équipe Neslihan Hoegy et Sophie Legall qui m'ont accompagné durant cette thèse. Je remercie aussi Laurent David, Perrine Paul-Gilloteaux et Magalie Feyeux pour leurs conseils et leur aide.

Je remercie Mme Jean, thérapeute avec qui j'ai appris à mieux gérer mon anxiété, qui a été exacerbée durant cette thèse. Je remercie ma famille : ma mère, mon père, mon frère, mes grand-parents et mes amis : Mokhtar, Ambre, Mathilde, Nathan, Paule, Dorian et Maxime pour avoir été à mes côtés pendant ces trois années. Enfin, je remercie Mouette et Aurore pour leur patience, leur amour et pour cette épaule fragile et solide à la fois qu'elles m'ont apportée.

SOMMAIRE

INTRODUCTION

Cadre général et objectifs. L'infertilité est un problème de santé mondial [72]. Le nombre de couples déclarant leur infertilité et s'adressant à des centres de procréation médicalement assistée (PMA) pour un bilan et un traitement de l'infertilité en Europe augmente de 8 à 9 % chaque année [38]. L'un des traitements les plus courants pour les couples infertiles est la Fécondation In Vitro (FIV). Elle consiste en une hyperstimulation ovarienne contrôlée, suivie d'un prélèvement d'ovules, d'une fécondation et d'une culture d'embryons pendant 2 à 6 jours dans des conditions environnementales contrôlées, conduisant au transfert intra-utérin ou à la congélation des embryons identifiés comme ayant un bon potentiel d'implantation par les embryologistes. L'efficacité clinique de la FIV est variable d'une région à l'autre, l'efficacité rapportée variant de 20 % à 40 %. La FIV est principalement entravée par les limites actuelles des méthodes d'évaluation de la qualité des embryons [35]. En effet, la principale méthode d'évaluation de la qualité des embryons est basée sur l'évaluation morphologique, qui consiste en une observation statique quotidienne au microscope. Bien qu'il existe un consensus pour l'évaluation morphologique du développement embryonnaire, cette méthode souffre encore d'un manque de pouvoir prédictif et d'une variabilité inter- et intra-opérateur [27, 139, 11].

Les incubateurs à imagerie Time-lapse ("Time-Lapse Imagery", TLI) ont fait leur apparition sur le marché de la FIV vers l'année 2010. Ils permettent un suivi continu du développement de l'embryon, en prenant des photos de chaque embryon à intervalles réguliers, pour finalement compiler une vidéo donnant un aperçu dynamique du développement embryonnaire in vitro. Cette technologie produit des conditions de culture très stables et permet d'annoter les événements du développement embryonnaire, appelés paramètres morphocinétiques (MC), tels que, par exemple, les divisions cellulaires ou la formation du blastocyste. Bien que plusieurs études aient rapporté une association entre les paramètres MC et le potentiel d'implantation, l'utilité clinique de la TLI reste débattue [117, 112, 8]. Néanmoins, la TLI semble toujours être la solution la plus prometteuse pour améliorer les méthodes d'évaluation de la qualité des embryons, et par la suite l'efficacité clinique de la FIV. En particulier, le volume élevé sans précédent d'images de haute qualité produites par les systèmes TLI peut être exploité à l'aide de réseaux de neurones profonds ("Deep Neural Networks", DNN).

Une limitation importante au développement de ces solutions est la nature opaque des modèles

proposés qui pose des problèmes notamment éthiques déjà soulevés par la communauté [4]. Mihdi et al. notent que de l'application de modèles non interprétables à la FIV résultent des problèmes de confiance, de responsabilité et de généralisations variables d'une population à l'autre, entre autres [4]. Par exemple, l'utilisation de modèles opaque peut entraîner un déficit de responsabilité. Étant donné que les biologistes ne peuvent pas comprendre comment un modèle prend une décision, il pourrait être déterminé qu'ils ne sauraient être responsables des décisions prises automatiquement et la responsabilité devrait alors incomber à un autre agent [4]. Une solution à ce problème est donc de déployer des modèles interprétables que l'utilisateur peut surveiller et comprendre afin d'intervenir lorsque le modèle n'utilise pas les bons indices par exemple. Dans la littérature, on trouve plusieurs types de méthodes d'explication pour améliorer l'interprétabilité des modèles. Par exemple, on trouve des méthodes qui permettent de comprendre le modèle dans sa globalité [105, 121], et des méthodes à base d'exemples [21, 107] ou de contre-exemples [52]. Dans le cadre de la classification d'image, on trouve principalement des explications sous la forme de *carte de saillance*, des cartes de chaleur qui indiquent les zones de l'image qui ont été importantes pour le modèle. Dans le cadre de cette thèse, c'est cette forme d'explication que nous étudions.

Contributions. Dans ce manuscrit, nous proposons plusieurs outils et modèles pour pallier l'opacité des DNN. Nos contributions s'établissent à différents niveaux :

— Nous avons développé une base de données, annotées par plusieurs experts, que nous avons rendu publique afin de compenser le manque de données permettant à la communauté de comparer les algorithmes développés et d'arriver à un consensus.
— Ensuite, un nouveau modèle d'attention artificielle spatiale est proposé. Pour rendre les décisions des réseaux plus transparentes et explicables, nous avons travaillé sur un nouveau mécanisme d'attention artificielle non paramétrique générant des cartes de saillance détaillées et interprétables, qui contraste avec les modèles paramétriques en basse résolution existants.
— Nous contribuons également à l'évaluation de l'interprétabilité en comparant cette proposition avec l'état de l'art de l'attention artificielle du point de vue de la fiabilité des cartes de saillance produites à l'aide de métriques objectives. Nous discutons des limites et difficultés conceptuelles posées par ces métriques et proposons des solutions partielles.

Plan de la thèse. La première partie consistera en une étude de mécanismes d'attention interprétables et de méthodes d'évaluation dans le cadre général de la classification d'image.

Dans le chapitre 1, nous décrivons le contexte scientifique de cette thèse. Nous introduisons les concepts fondamentaux d'explicabilité et d'interprétabilité et nous situons dans le cadre des approches de saillance en vision par ordinateur. Nous détaillons ensuite les deux grandes familles d'approches pour générer des cartes de saillance. Les méthodes de la première famille sont appelées méthodes post-hoc et calculent des explications visuelles après l'inférence. Elles se distinguent en trois groupes : les méthodes par perturbation, par pondération des cartes de caractéristiques et par rétropropagation vers l'image d'entrée. Au contraire, la famille des modèles d'attention propose de générer une carte d'attention pendant l'inférence. Ces modèles se divisent en trois types d'approches : l'attention convolutionelle, l'attention prototypiques et l'attention non paramétrique. Nous évoquons ensuite les méthodologies existantes pour évaluer leur interprétabilité. Nous détaillerons particulièrement les *métriques de fiabilité*, des métriques objectives pour évaluer la fiabilité des cartes de saillance, car nous les utiliserons dans plusieurs chapitres de ce manuscrit.

Le chapitre 2 est concentré sur les métriques de fiabilité. D'abord, nous analysons les limites de deux métriques appelées "Aire sous la courbe en supression" ("Deletion Area Under Curve", DAUC) et "Aire sous la courbe en insertion" ("Insertion Area Under Curve", IAUC) : premièrement, nous montrons qu'elles évaluent la qualité des explications avec des inférences exécutées sur des images hors de la distribution d'entraînement du modèle. Deuxièmement, ces métriques ignorent les scores des cartes de saillances et ne tiennent compte que de l'ordre des pixels. Afin de pallier ce problème, nous proposons deux métriques nouvelles et concluons par une discussion sur ces métriques et leur rôle potentiel au sein d'une étude utilisateur.

Dans le chapitre 3, nous introduisons un modèle d'attention non paramétrique appelé BR-NPA générant des cartes d'attention en haute résolution fiables et détaillées. En utilisant diverses évaluations incluant notamment des métriques de fiabilité, nous avons montré que BR-NPA génère des cartes d'attention fiables tout en proposant de meilleures performances de classification que les autres modèles interprétables et méthodes post-hoc étudiées. De plus l'observation des cartes de BR-NPA montrent que ce modèle focalise précisément son attention sur l'image, permettant de voir qu'il utilise des indices visuels pertinents pour la tâche à accomplir.

Dans la partie II, nous appliquons ces architectures interprétables ainsi que les méthodes d'évaluations développées à un problème de classification d'image issues de la FIV. Dans le chapitre 4, nous introduisons les vidéos time-lapse d'embryons ainsi que la tâche de prédiction des paramètres MC. De plus, nous constatons deux limites des travaux précédents proposant d'appliquer des DNN à la FIV. Premièrement, les données utilisées sont privées et les modèles ne sont entraînés à modéliser qu'un nombre réduit de stades de développement embryonnaire.

Deuxièmement, seulement une seule étude à notre connaissance a déjà proposé d'utiliser une architecture interprétable pour les images d'embryon time-lapse.

Dans le chapitre 5, nous introduisons donc une base de référence constituée d'un grand nombre de vidéos time-lapse d'embryon complètes accompagnées d'annotations détaillées. Les annotations recouvrent tous les stades de développement de l'embryon du premier au cinquième jour et permettent d'enrichir les prédictions faites par le modèle en comparaison aux travaux précédents. Nous observons le gain de performance apporté par les DNN conçus pour le format vidéo par rapport à l'approche Ad-Hoc précédente, confirmant l'intérêt d'un jeu de données constitués de vidéos complètes suffisamment grand pour l'apprentissage profond (AP).

Le chapitre 6 compare la fiabilité des cartes de saillance générées par des méthodes d'explications post-hoc génériques avec les cartes générées par des modèles d'attention interprétables. Nous montrons que le type de métrique utilisé impacte largement le type d'approche favorisé (modèle d'attention ou méthode post-hoc). Ensuite sont identifiées les approches les plus fiables selon chaque type de métrique, à savoir BR-NPA et Score-CAM. À l'aide d'une comparaison qualitative des cartes produites par ces deux approches, nous montrons que la résolution élevée des cartes de BR-NPA est un facteur déterminant permettant d'identifier les indices visuels utilisés par le modèle.

PREMIÈRE PARTIE

Approches de saillances

CHAPITRE 1

ÉTAT DE L'ART DES APPROCHES DE SAILLANCE

Dans ce chapitre nous introduisons d'abord les concepts d'interprétabilité et d'explication puis détaillons la manière dont ils se traduisent souvent en vision par ordinateur à travers les approches de saillance.

1.1 L'interprétabilité

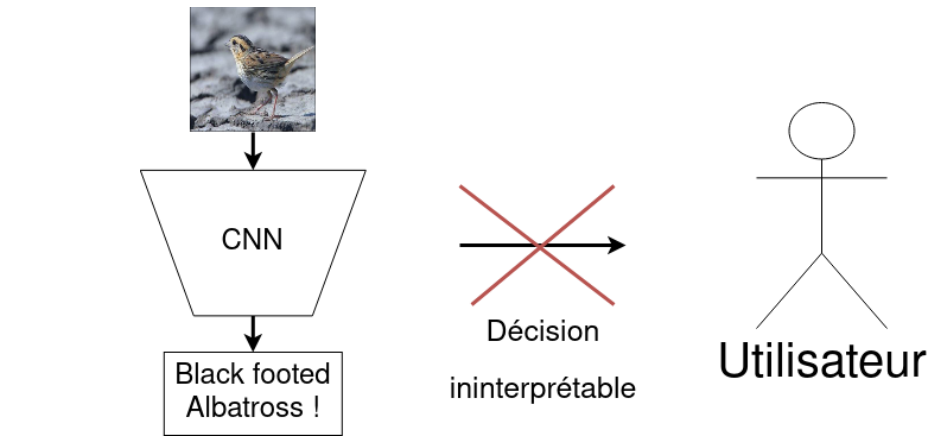
L'interprétabilité (ou "explicabilité") d'un système est sa capacité à faire comprendre ses décisions à un humain notamment à l'aide d'explications. Selon la CNIL, c'est "la capacité de mettre en relation et de rendre compréhensible les éléments pris en compte par le système d'IA pour la production d'un résultat" [37].

L'étude de ce type de système est un domaine qui a connu une renaissance avec l'arrivée de l'apprentissage profond. En effet, les excellentes performances des DNN sur de nombreuses applications fait qu'il est désormais envisageable de les déployer dans des domaines comme la médecine ou la justice. Cependant, la complexité de ces modèles est un frein important à ce déploiement. Étant donné les enjeux de ces domaines, il n'est pas envisageable de laisser un algorithme prendre des décisions seul, car ces modèles sont en fait peu robustes et les décisions produites peuvent refléter des biais indésirables des données sur lesquelles ils ont été entraînés. Une erreur de jugement dans ces domaines a des conséquences graves et il est donc impératif qu'au moins un expert vérifie la validité des décisions du modèle. Cette vérification est difficile en pratique avec un DNN standard car ce type de modèle comporte des millions de paramètres et, contrairement à un modèle de régression linéaire où il n'y a que quelques coefficients, il n'est pas possible de comprendre directement comment le modèle a pris sa décision. Une nouvelle branche de l'étude de l'apprentissage profond a donc émergé ces dernières années et porte son attention sur l'explicabilité de ces modèles et propose des méthodes pour améliorer et évaluer cette dernière. C'est dans ce champ d'étude que s'inscrit ce travail et plus particulièrement

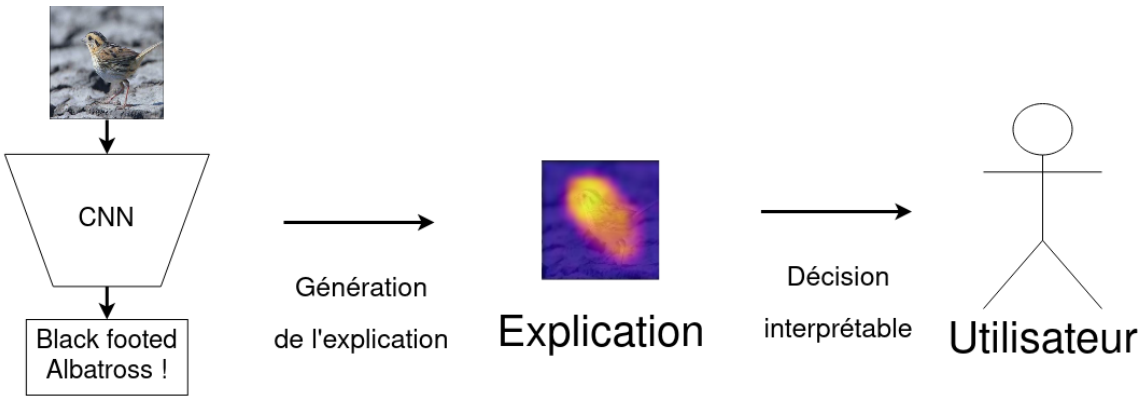
à son application à la vision par ordinateur. Ce champ étant encore jeune, la définition de la notion même d'interprétabilité est mal définie et la manière de l'évaluer ne fait pas l'objet d'un consensus. Par exemple, on pourrait argumenter que cette propriété ne devrait pas être binaire et est probablement multidimensionnelle. En effet, on peut argumenter que l'explication fournie par un modèle doit s'adapter au public visé et doit notamment dépendre du niveau d'expertise des utilisateurs dans la tâche que le modèle automatise. On peut donc envisager qu'un modèle soit interprétable aux yeux d'un certain public et peu aux yeux d'un autre. Aussi, comme nous le verrons dans les chapitres suivants, l'évaluation de cette propriété est difficile pour plusieurs raisons : premièrement par l'absence de consensus sur la définition, deuxièmement par le coût des expériences utilisateurs et le manque d'expertise pour les réaliser et troisièmement par la diversité des besoins en interprétabilité.

Malgré ces difficultés fondamentales, de nombreuses approches pour améliorer l'interprétabilité des DNN ont été proposées, parmi lesquelles on distingue les approches *locales* et les approches *globales*. Une méthode globale éclaire le comportement d'un modèle sur un jeu de données entier en montrant par exemple quelles sont les caractéristiques auxquelles le modèle accorde un poids important [121] ou dans quels domaines de l'espace d'entrée le modèle a une performance limitée [105]. Au contraire, une méthode locale cherche à expliquer une décision sur une entrée spécifique. Elle montre les éléments de l'entrée qui ont été importants dans la décision, dans l'optique d'aider l'utilisateur à mieux comprendre pourquoi le modèle a produit une certaine décision et non pas une autre. Elle sert d'intermédiaire entre le modèle et l'utilisateur afin de rendre interprétable la décision prise par le modèle, comme illustré en figure 1.1.

Les explications globales s'adressent aux experts du modèle et sont efficaces pour un audit ou une évaluation globale du modèle par exemple. Pour faciliter la collaboration entre l'humain et le modèle dans le cadre d'un usage routinier, il est préférable d'utiliser des explications locales car elles permettent de gagner la confiance de l'utilisateur sur des entrées précises à traiter [105]. Étant donné que le but du projet est d'aider les biologistes dans le cadre de leur travail quotidien, nous nous focalisons exclusivement sur les explications locales. En vision par ordinateur, une large majorité des approches d'explicabilité et d'interprétabilité sont locales. On trouve notamment les explications à l'aide d'exemples [21, 107] ou de contre exemples [52] ainsi que les méthodes de *saillance*. Ces méthodes consistent à générer des cartes de chaleur indiquant les zones de l'image ayant joué un rôle important dans la décision du modèle. Ces cartes de chaleurs sont appelées *cartes de saillance*. Dans le cadre de cette thèse, nous n'utiliserons que des méthodes de saillance, étant donné la popularité de ces approches. Nous verrons d'abord les différentes approches pour calculer des cartes de saillance, puis nous verrons les méthodologies



(a)



(b)

FIGURE 1.1 – Illustration du rôle de l'explication. (a) Dans le cas d'un modèle boite noire sans explication, la décision n'est pas interprétable par l'utilisateur. (b) Une explication sert d'intermédiaire entre l'utilisateur et le modèle afin de rendre la décision interprétable.

existantes pour les évaluer. Nous détaillerons particulièrement les *métriques de fiabilité*, des métriques objectives pour évaluer la fiabilité des cartes de saillance, car nous les utiliserons dans plusieurs chapitres de ce manuscrit.

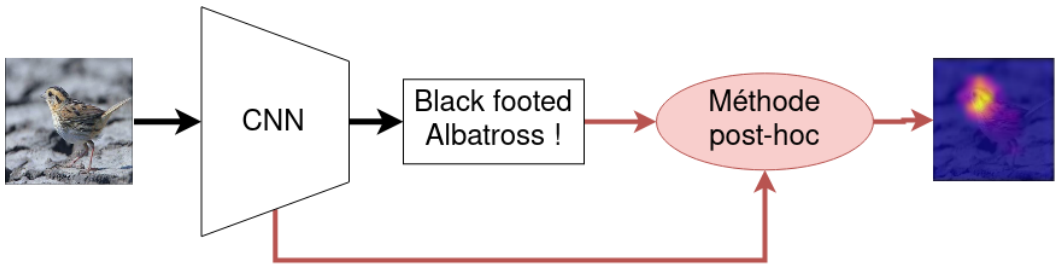
1.2 Les approches de saillance en vision par ordinateur

Face à la nature en boite noire des modèles d'apprentissage profonds de classification d'images, deux approches ont émergé de la littérature pour générer des explications basées sur des cartes de saillance. La première propose des méthodes permettant d'expliquer les décisions de n'importe quel modèle, elles sont dites *post-hoc*. La deuxième tendance consiste à intégrer des éléments censés améliorer l'interprétabilité à une architecture, tels qu'une couche d'attention. Un trait commun de ces deux approches est la génération de cartes de saillances qui sont des cartes de chaleur mettant en valeur les éléments qui ont été importants dans la décision du modèle. Pour les méthodes post-hoc, ces cartes sont obtenues par un calcul supplémentaire à l'inférence alors que les architectures interprétables génèrent la carte pendant la phase d'inférence, comme illustré en figure 1.2. D'autres approches existent pour générer des explications de modèle de classification d'image comme [82] qui propose d'extraire des concepts compréhensibles par un humain ou [170] qui trouve des parties similaires des images d'entraînement à l'image d'entrée. Cependant, nous n'incluons donc que des approches de saillance ici pour deux raisons. Premièrement, la littérature de vision par ordinateur interprétable se concentre surtout sur les approches de saillance et deuxièmement, il est difficile de comparer des approches de saillance avec d'autres type d'approches.

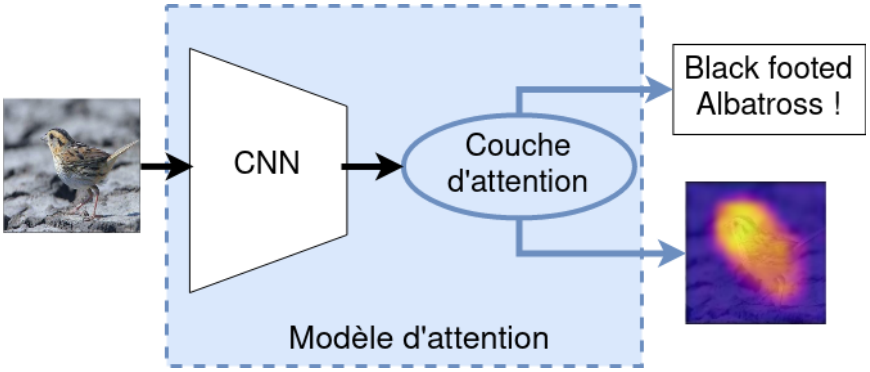
Premièrement nous étudions les méthodes génériques dite "post-hoc" qui servent à expliquer les décisions prises par n'importe quel modèle de classification visuelle. Deuxièmement, nous aborderons les architectures intégrant une couches d'attention. Nous conclurons avec les limites de ces approches.

1.2.1 Méthodes d'explication génériques

On distingue trois types d'approches génériques pour produire des explications d'un modèle sans avoir à le réentraîner : les approches par pondération des cartes de caractéristiques (que l'on appellera aussi *cartes de traits*), les approches par rétropropagation vers l'image d'entrée et les approches par perturbation de l'image d'entrée, comme illustré en figure 1.3.



(a) Méthodes post-hoc



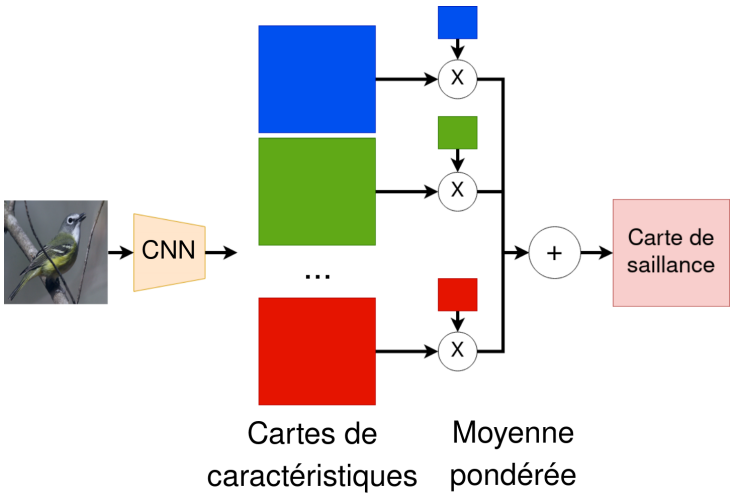
(b) Modèles d'attention

FIGURE 1.2 – Les deux approches pour produire des cartes de saillance.

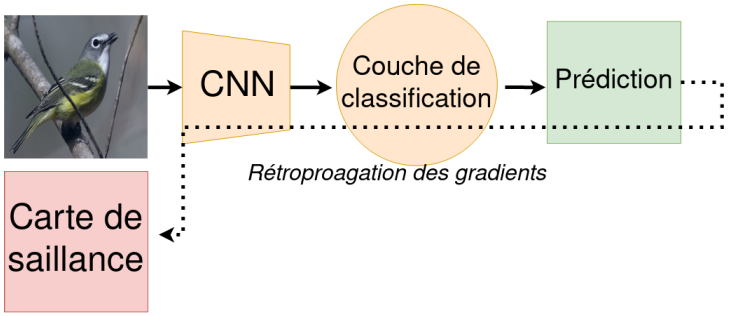
Notations. Soit $x \in \mathbb{R}^{H \times W \times 3}$ l'image en couleur passée au modèle, $y_c(x)$ est le score de la classe à expliquer. On note également $(i, j) \in [0, ..., H] \times [0, ..., W]$ la position spatiale au niveau de l'image d'entrée et $(i', j') \in [0, ..., H'] \times [0, ..., W']$ la position spatiale au niveau des cartes de caractéristiques de la dernière couche.

Approches par perturbation de l'image d'entrée. Ces méthodes consistent à perturber l'image d'entrée pour déterminer quelle est la contribution de chaque partie au score y_c. Dans cette catégorie on trouve par exemple la méthode appelée "Randomized Input Sampling for Explanation" (RISE) [115] (échantillonnage aléatoire d'entrées pour l'explication) qui a été proposée par Petsiuk et al.. Cette méthode consiste d'abord à découper l'image en un grille rectangulaire $H' \times W'$ et à calculer Q masques binaires aléatoires $m^q \in \{0, 1\}^{H' \times W'}$ où $m^q_{i'j'} \sim$ Bernoulli(0.5) et $q \in [1, ..., Q]$. Ensuite sont calculés Q inférences en appliquant un masque différent sur l'image à chaque fois :

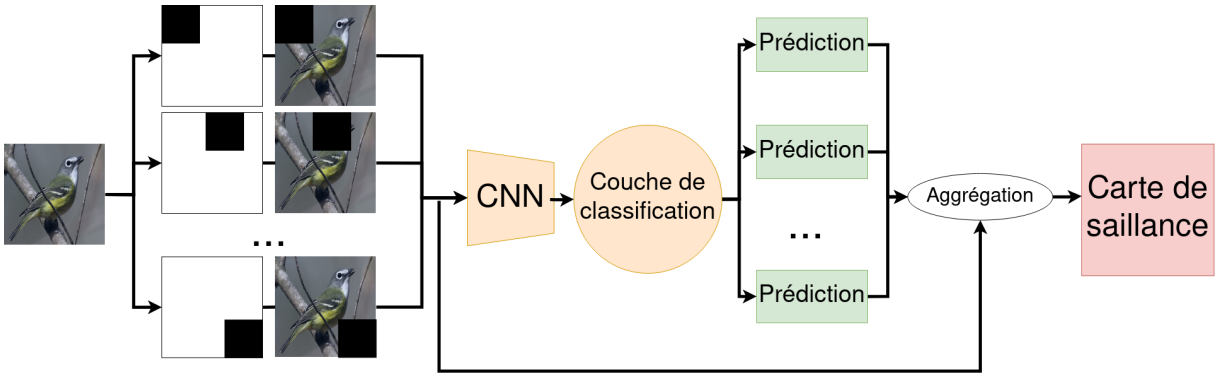
$$c_q = y_c(x \cdot interpol(m^q)), \tag{1.1}$$



(a) Approche par pondération des cartes de caractéristiques.



(b) Approche par rétropropagation du gradient vers l'image d'entrée.



(c) Approche par perturbation de l'image d'entrée.

FIGURE 1.3 – Illustrations des trois types de méthodes d'explications traités dans ce manuscrit.

où l'opérateur $interpol$ augmente la résolution du masque avec une interpolation par plus proche voisins afin de correspondre à la résolution de l'image d'entrée. Cette procédure consistant à masquer l'image puis à mesurer la variation du score est répétée en pratique plusieurs milliers de fois pour estimer quelles sont les zones de l'image qui, quand elles sont masquées, induisent la plus forte chute du score, et donc sont les plus importantes pour la décision, comme illustré en figure 1.4. La carte de saillance finale est calculée comme suit :

$$S = \frac{1}{Q} \sum_{q} c_q \times m_q \tag{1.2}$$

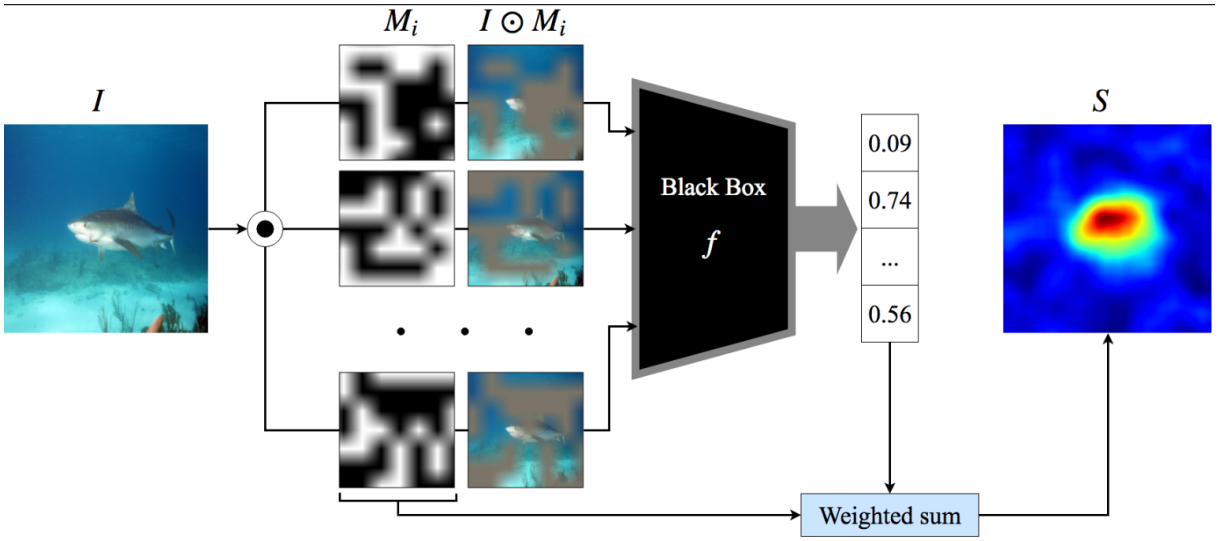


FIGURE 1.4 – Illustration de la méthode RISE. Des masques aléatoires sont appliqués sur l'image d'entrée ce qui permet d'obtenir de nouveaux scores pour la classe initialement prédite. La carte de saillance est calculée par une moyenne pondérée des masques où les poids sont les scores produits par le modèle. Extrait de [115].

La méthode LIME [121] fonctionne de façon similaire exceptée qu'elle ne découpe pas l'image en une grille régulière mais en super-pixels et que le poids de chaque super-pixels correspond au poids d'un modèle linéaire G qui approxime le modèle F localement. Soit x l'image d'entrée et x' une version simplifiée de x. Plus précisément x' est un vecteur binaire où $x'i$ indique si le i-ième super-pixel est présent dans x ($x'_i = 1$) ou s'il est masqué avec des pixels gris ($x'_i = 0$). On note $G(x')$ la prédiction du modèle linéaire en masquant les super-pixels indiqués par x'. Le modèle F est approximé comme suit :

$$G(x') = \phi_0 + \sum_{i}^{P} \phi_i x'_i, \tag{1.3}$$

où les $\{\phi_i\}_i$ sont les poids de chaque super-pixels. Les auteurs définissent la fonction de coût suivante :

$$Loss = \Omega(G) + \sum_{\hat{x} \in Z} \pi(\hat{x}, x) \left(y_c(\hat{x}) - G(\hat{x}') \right), \tag{1.4}$$

où X est un ensemble obtenu en échantillonnant le voisinage de x, $\pi(x, x') = \exp(-||x - x'||_2^2/\sigma^2)$ est une fonction de similarité permettant de donner un plus grand poids aux voisins de z qui sont les plus proches et Ω est un terme de régularisation pour diminuer la complexité dans l'explication. Un échantillon $\hat{x}$ est obtenu en masquant aléatoirement des super-pixels de l'image ce qui revient à changer les valeurs de x'. En pratique les auteurs de LIME propose d'utiliser $|X| = 15\ 000$ échantillons par image à expliquer. Afin de réduire la complexité de l'explication, le terme de régularisation suivant est choisi :

$$\Omega(G) = \begin{cases} \infty \text{ si } ||\phi||_0 > C \\ 0 \text{ sinon,} \end{cases} \tag{1.5}$$

où $||\phi||_0$ est le nombre de coefficients dont la valeur est non-nulle. En pratique, pour respecter cette contrainte, les auteurs choisissent les C super-pixels les plus importants avec l'algorithme LASSO [36] et détermine leur poids ϕ_i avec une régression en moindres carrés. Les autres super-pixels se voient attribuer un poids nul. Enfin, Lundberg et al. proposent d'utiliser les valeurs de Shapley comme solution à l'équation (1.3) avec une méthode appelée SHAP [98]. Les auteurs montrent que les valeurs de Shapley sont les seules solution de l'équation (1.3) permettant de garantir des propriétés intéressantes telles que la fidélité locale ou la cohérence : si le modèle change tel que la contribution d'un super-pixel augmente ou reste identique, le poids du super-pixel ne décroîtra pas. Soit $\mathcal{P} = [0, ..., P]$ l'ensemble des indexs des super-pixels où P est le nombre de super-pixel. Soit p un sous-ensembles de $\mathcal{P}$. On note y_c^p un modèle entraîné avec les super-pixels dont les index sont inclus dans p. De la même manière, on note x_p l'image x dont seul les super-pixels indiqués par p ne sont pas masqués. La valeur de Shapley d'un super-pixel d'index i est calculée par l'espérance de la différence entre $y_c(x)$ et $y_c^{p \cup \{i\}}(x_{p \cup \{i\}})$ où S est choisi aléatoirement parmi tous les sous-ensemble de $\mathcal{P} \setminus \{i\}$. Les poids ϕ_i sont calculés comme suit :

$$\phi_i = \sum_{p \subseteq \mathcal{P} \setminus \{i\}} \frac{|p|!(P - |p| - 1)!}{P!} \left(y_c^{p \cup \{i\}}(x_{p \cup \{i\}}) - y_c^p(x_p) \right), \tag{1.6}$$

Le calcul de l'équation (1.6) est cependant irréalisable parce que chaque terme de la somme implique d'entraîner un nouveau modèle et que le nombre de termes de la somme est exponentiel-

lement grand avec le nombre de super-pixel. Afin de ne pas devoir entraîner un modèle par terme de la somme, les auteurs utilisent le même modèle à chaque terme en masquant simplement les super-pixels à ignorer dans l'image. Aussi, pour réduire le nombre de terme, des voisins de x sont échantillonnés et les coefficients sont obtenus en optimisant une fonction à la manière de LIME. Cependant, afin de garder les propriétés évoquées plus haut, la fonction de similarité π, la régularisation Ω et la fonction de coût L utilisées sont différentes :

$$\pi(x, z) = \frac{P - 1}{\binom{P}{|z'|}|z'|(P - |z'|)} \tag{1.7}$$

$$\Omega(G) = 0 \tag{1.8}$$

$$Loss = \sum_{\hat{z} \in Z} \pi(\hat{z}, z) \left(y_c(\hat{z}) - G(\hat{z}')\right)^2, \tag{1.9}$$

où $|z'|$ est le nombre de super-pixel non masqué dans z.

Une limite de ces méthodes est le nombre d'inférence à exécuter pour avoir une bonne estimation de la carte de saillance qui est exponentiel avec la résolution de la carte/le nombre de super-pixels. Par exemple, Petsiuk et al. exécutent $M = 8000$ inférences par image de taille standard 224×224 pour obtenir des cartes avec la méthode RISE en résolution 7×7 avec une colonne vertébrale [1] standard ResNet-50 [115]. De façon similaire, les auteurs de LIME utilisent 15000 inférences par image à expliquer. Contrairement aux deux autres familles que nous évoquons ensuite, ces méthodes peuvent s'appliquer sur tout type de classifieur d'image et n'est donc pas restreint aux CNN ou même aux DNN. Cependant, cet intérêt est limité en pratique actuellement car la plupart des modèles utilisés en vision par ordinateur aujourd'hui sont des CNN ou des DNN.

Approches par pondération des cartes de caractéristiques. Une autre tendance pour l'explication des modèles profonds se base sur la pondération des cartes de caractéristiques produites à la dernière couche du CNN pour expliquer la classe prédite. Ces méthodes produisent des cartes de saillance à la résolution des cartes de caractéristiques de la dernière couche, i.e. $S \in \mathbb{R}^{H' \times W'}$. Un travail fondamental de cette catégorie est la méthode appelée "carte d'activation de classe" ("Class Activation Map", CAM) [176] et a été proposée pour visualiser les zones qui ont le plus contribué à la prédiction d'une classe spécifique. Pour cela, les auteurs agrègent les cartes de caractéristiques de la dernière couche en les pondérant par le poids qu'elles ont sur le score de la classe à expliquer, comme illustré en figure 1.5.

1. Traduction de l'anglais "backbone".

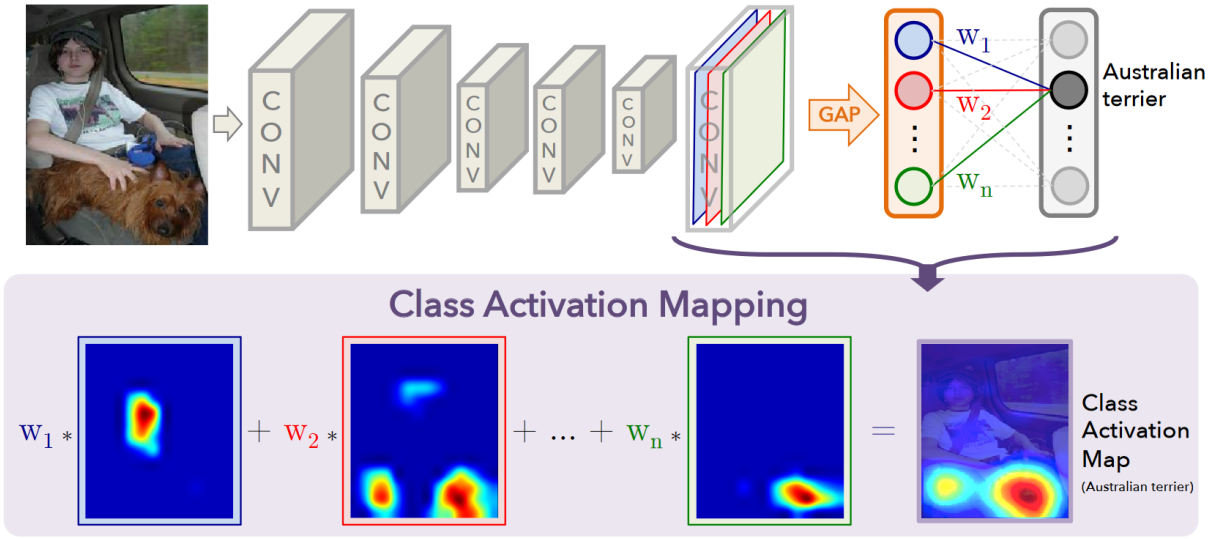


FIGURE 1.5 – Illustration de la méthode CAM. Extrait de [176]. La carte de saillance est obtenue avec une moyenne pondérée des cartes de caractéristiques de la dernière couche de convolution, en utilisant les poids de la couche de classification.

Soit $f^k \in \mathbb{R}^{H' \times W'}$, c, et w_{kc} la k-ième carte de caractéristiques de la dernière couche, l'index de la classe à expliquer et le poids de la couche linéaire de classification connectant f_k et le score de la classe c. La carte de saillance est calculée comme suit :

$$CAM(x) = \sum_k w_{kc} \times f^k(x) \tag{1.10}$$

Une des limites de cette méthode est qu'elle nécessite que la couche de classification soit connectée directement aux cartes de caractéristiques et ne peut donc pas être appliqué à des architectures de type VGG par exemple, où il y a plusieurs couches denses entre les cartes et les scores de classification. Elle ne peut pas non plus être appliquée à des modèles conçus pour d'autre tâches que la classification comme la description d'images ou la réponse à des questions visuelles ("Visual Question Answering", VQA). Grad-CAM [127] résout ce problème en généralisant CAM pour être applicable à toute les architectures où les scores de classes sont calculés de façon différentiables, et non plus seulement de façon linéaire, à partir des cartes de caractéristiques. Pour cela, Grad-CAM calcule les gradients du score de la classe à expliquer par rapport aux cartes de caractéristiques pour identifier les cartes qui ont le plus contribué à la décision :

$$\text{Grad-CAM}(x) = \sum_k w^g_{kc} \times f^k(x), \tag{1.11}$$

où w_{kc}^{g} est défini comme suit :

$$w_{kc}^{g} = \frac{1}{H'W'} \sum_{i'=0}^{H'} \sum_{j'=0}^{W'} \frac{\partial y_c(x)}{\partial f_{i'j'}^{k}(x)} \tag{1.12}$$

Notez que lors du calcul des w_{kc}^{g}, les dérivées partielles sont toutes agrégées avec le même poids [19]. Chattopadhyay et al. argumentent qu'en conséquence, la localisation ne correspond pas à l'objet entier, mais à des parties de celui-ci. Aussi, s'il y a plusieurs occurrences d'un même objet, les cartes de saillance de Grad-CAM ne les met pas toutes en valeur. Cela peut être considéré comme un problème étant donné que cette situation est courante hors du domaine de la classification d'images mono-label par exemple. Chattopadhyay et al ont donc proposé Grad-CAM++ [19], une variante de Grad-CAM où chaque dérivée partielle se voit assigner un poids différent. Comme avec CAM, on se situe dans un cadre où il n'y a qu'une seule couche dense et on a donc :

$$w_{kc} = \frac{\partial y_c(x)}{\partial f_{i'j'}^{k}(x)} \tag{1.13}$$

Cependant, cette variante de Grad-CAM propose de ne pas utiliser l'équation (1.13) pour calculer les poids des cartes mais l'équation suivante :

$$w_{kc}^{g++} = \frac{1}{H'W'} \sum_{i'=0}^{H'} \sum_{j'=0}^{W'} a_{ij}^{kc} ReLU\left(\frac{\partial y_c(x)}{\partial f_{i'j'}^{k}(x)}\right) \tag{1.14}$$

On note deux modifications par rapport à Grad-CAM. Premièrement, une fonction d'activation $ReLU(\cdot)$ est appliquée sur les dérivées partielles afin d'ignorer les pixels qui influent de façon négative sur le score y_c. Deuxièmement, les dérivées partielles de l'équation (1.12) sont multipliées par des poids a_{ij}^{kc}. Les auteurs obtiennent une forme close pour ces poids en faisant l'hypothèse que $w_{kc} = w_{kc}^{g++}$ et en combinant l'équation (1.14) avec l'équation liant y_c et les f^k :

$$y_c(x) = \sum_k w_{kc} \sum_{ij} f_{ij}^{k}(x) \tag{1.15}$$

Une fois les poids calculés, les cartes sont agrégées de la même façon que le font Grad-CAM et CAM :

$$\text{Grad-CAM++}(x) = \sum_k w_{kc}^{g++} \times f^k(x) \tag{1.16}$$

Notez qu'utiliser Grad-CAM++ implique qu'il n'y a qu'une couche dense et la classe d'architecture est donc la même que celle de CAM, et est restreinte par rapport à celle de Grad-CAM. On

remarque aussi que la méthodologie utilisée par les auteurs est critiquable. En effet, l'hypothèse disant que w_{kc} est égal à w_{kc}^{g++} est fausse : w_{kc} est le paramètre de la couche linéaire appris pendant l'entraînement liant f^k et y_c et w_{kc}^{g++} est une somme de moyenne pondérée de dérivée partielle positives. De façon générale, ces quantités ne sont pas égales.

Wang et al. argumentent que les approches basées sur des calculs de gradients telles que Grad-CAM et Grad-CAM++ ont deux problèmes. Premièrement, les phénomènes de saturation/évanescence de gradients causerait une apparence bruitée des cartes de saillance et deuxièmement, ces approches seraient susceptibles d'accorder un poids trop important à certaines cartes de caractéristiques [155]. Pour ne pas souffrir de ces problèmes, Wang et al. proposent Score-CAM, une méthode qui attribue un poids à chaque carte de caractéristiques en fonction de l'augmentation du score observé en masquant les zones de l'image d'entrée qui n'activent pas la carte.

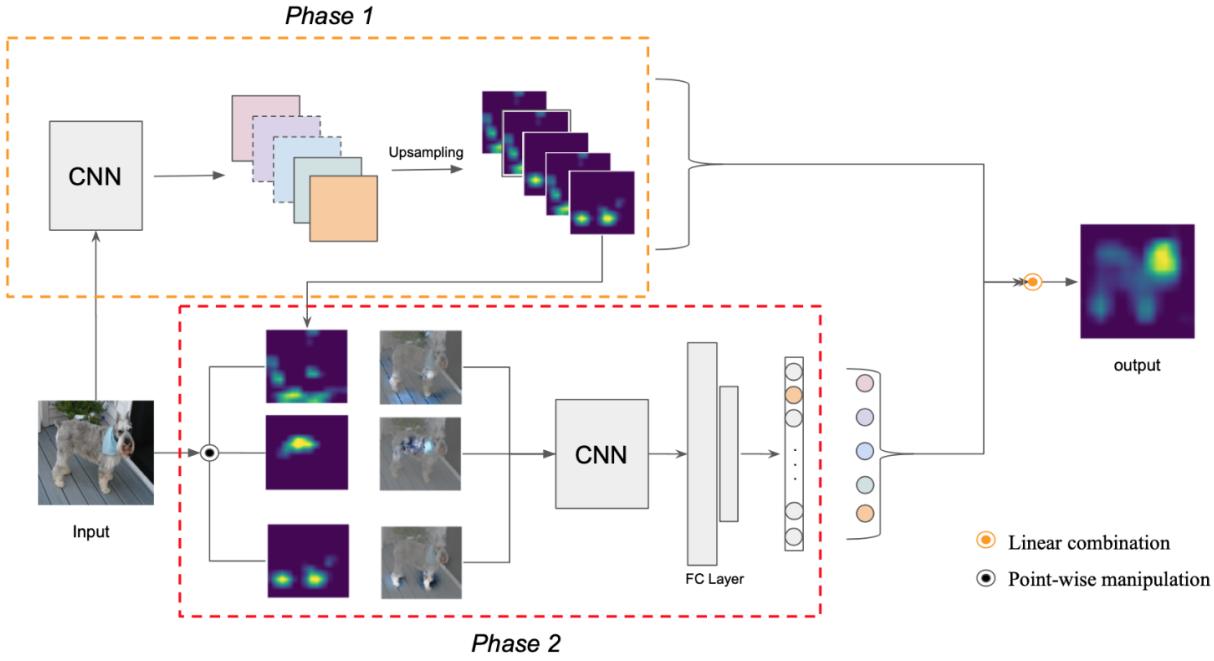


FIGURE 1.6 – Illustration de la méthode Score-CAM. L'image est masquée successivement par chaque carte de caractéristique afin d'examiner l'impact de chaque zone sur le score de la classe. La carte de saillance est obtenue avec une moyenne pondérée des cartes où les poids correspondent à la variation du score. Extrait de [155].

La figure 1.6 illustre le méthode Score-CAM. Comme les méthodes évoquées précédemment,

Score-CAM se formule ainsi :

$$\text{Score-CAM}(x) = \sum_k w_{kc}^s \times f^k(x) \tag{1.17}$$

La différence avec les méthodes précédentes intervient dans la formulation des poids w_{kc}^s,. Ceux-ci sont définis comme la variation du score obtenues après avoir masqué les zones de l'image n'activant pas f_k :

$$w_{kc}^s = y_c(x) - y_c(x \cdot Norm(Upsample(f^k))), \tag{1.18}$$

où $Upsample$ calcule une interpolation de f_k à la résolution de x et $Norm$ est un opérateur qui normalise la carte entre 0 et 1 comme suite : $Norm(x) = \frac{x - x_{min}}{x_{max} - x_{min}}$. Une méthode similaire à Score-CAM nommée Ablation-CAM été introduite en parallèle par Desai et al. [30] avec des motivations proches de celles de Score-CAM. Ablation-CAM évalue l'importance d'une carte en empêchant la décision d'exploiter les informations qu'elle contient et en mesurant la variation relative du score y_c. Cependant, au lieu de masquer l'image d'entrée, Ablation-CAM supprime la carte du calcul de y_c et mesure la chute du score. Plus précisément les poids w_{kc}^a des cartes sont calculés comme suit :

$$w_{kc}^a = \frac{y_c(x) - y_c^{\setminus k}(x)}{y_c(x)}, \tag{1.19}$$

où $y_c^{\setminus k}$ est le score obtenu en supprimant f^k du calcul de y_c. Afin de ne mettre en valeur que les pixels dont la présence augmente le score, une fonction d'activation ReLU est appliquée après agrégation des cartes :

$$\text{Ablation-CAM}(x) = ReLU(\sum_k w_{kc}^a \times f^k(x)) \tag{1.20}$$

Il faut noter que ni Wang et al. ni Desai et al. ne démontrent si les deux problèmes ci-dessus existent vraiment chez Grad-CAM/Grad-CAM++ ou s'ils sont absent avec Score-CAM et Ablation-CAM, et on peut donc s'interroger sur la pertinence des motivations de ces méthodes.

Récemment, Jung et al. [77] ont proposé LIFT-CAM en remarquant que les valeurs de Shapley peuvent être utilisées pour évaluer l'importance des cartes de caractéristiques et suggèrent de les approximer avec l'algorithme DeepLIFT [130] pour obtenir une carte de saillance, comme illustré en figure 1.7. Cette idée est similaire à la méthode SHAP mais au lieu d'approximer les valeurs de Shapley de différentes parties de l'entrée, Jung et al. se placent dans le cadre des méthodes dérivant de CAM et approxime les valeurs de Shapley des différentes cartes de caractéristiques. Soit $\mathcal{K} = [1, ..., K]$ l'ensemble des index des cartes de caractéristiques avec

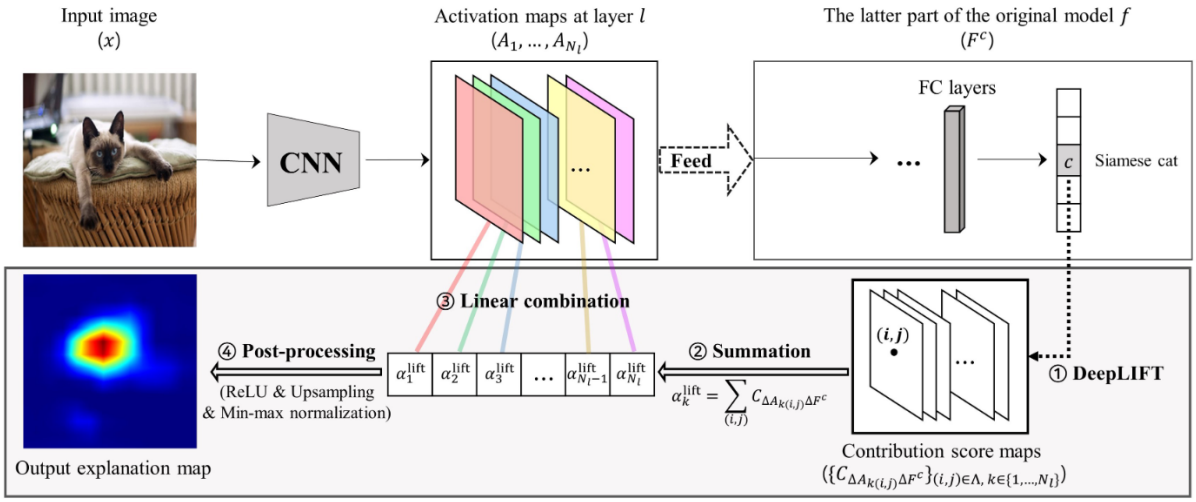


FIGURE 1.7 – Illustration de la méthode LIFT-CAM. Les poids des cartes de caractéristiques sont calculés avec l'algorithme DeepLIFT afin d'approcher les valeurs de Shapley. Extrait de [77].

K le nombre total de cartes. Soit κ un sous-ensemble de $\mathcal{K}$. On note également $y_c^{\kappa}(x)$ le score produit par le modèle en ignorant les cartes dont l'index est inclus dans κ. Les poids des cartes sont calculés avec une formule similaire à celle utilisée pour SHAP :

$$w_{kc}^{shap} = \sum_{\kappa \subseteq \mathcal{K} \setminus \{k\}} \frac{|\kappa|!(K - |\kappa| - 1)!}{K!} \left(y_c^{\kappa \cup \{k\}}(x) - y_c^{\mathcal{K}}(x) \right) \tag{1.21}$$

Afin de rendre ce calcul possible, les poids sont approximés en s'inspirant de l'algorithme DeepLIFT [130] :

$$w_{kc}^{lift-cam} = \sum_{ij} C_{\Delta f_{ij}^k \Delta y_c}, \tag{1.22}$$

où $C_{\Delta f_{ij}^k \Delta y_c}$ est l'impact de f_{ij}^k sur le score y_c. Les termes $C_{\Delta f_{ij}^k \Delta y_c}$ sont obtenus à partir de l'équation de "somme vers différence" postulé par les auteurs de DeepLIFT. Cette méthode d'explication sera détaillée plus tard mais on peut déjà dire qu'intuitivement, cette équation postule que chaque activation des cartes de caractéristiques de la dernière couche contribue à une partie de la différence $y_c(x) - y_c(\tilde{x})$ où où $\tilde{x} \in \mathbb{R}^{H \times W \times 3}$ est une image dite de *référence*. Il n'y a pas de définition formelle d'une image de référence dans la littérature mais de façon générale c'est une image qui doit représenter l'absence d'objet à classifier [142]. Ici, la référence est une image dont tous les pixels ont une valeur nulle. Dans le cadre de LIFT-CAM, l'équation "somme

vers différence" est la suivante :

$$\sum_{k=1}^{K} \sum_{ij} C_{\Delta f_{ij}^k \Delta y_c} = y_c(x) - y_c(\tilde{x}), \tag{1.23}$$

Les auteurs de DeepLIFT démontrent l'existence de plusieurs règles de calcul, analogues à celles de la rétropropagation, qui permettent notamment de calculer l'impact des activations des cartes sur la différence $y_c(x) - y_c(\tilde{x})$. Les auteurs ne proposent pas d'argument théorique montrant qu'utiliser les coefficients de DeepLIFT permet d'approximer les coefficients de Shapley. Cependant, ils montrent empiriquement que parmi plusieurs autres méthodes d'explications, LIFT-CAM est celle qui produit les poids se rapprochant le plus des vrais coefficient de Shapley, où ceux-ci sont estimés grâce à une approximation de l'équation (1.21) utilisant 10000 échantillons.

Pour donner un cadre théorique clair et suffisant, Fu et al. ont également proposé XGrad-CAM [42], une méthode d'explication basée sur des propriétés désirables que les auteurs proposent : la *sensibilité* et la *conservation*. La propriété de sensibilité propose que le poids w_{kc}^{χ} attribué à une carte soit égal à la variation du score lorsque l'on supprime f^k du calcul de y_c. Formellement, l'égalité suivante doit être vérifiée :

$$y_c(x) - y_c^{\setminus k}(x) = w_{kc}^{\chi} \sum_{ij} f_{ij}^k(x), \tag{1.24}$$

où $y_c^{\setminus k}$ est le score obtenu après avoir remplacé les activations de f^k par des valeurs nulles. La propriétés de conservation propose que le score doit s'expliquer exclusivement par la contribution des cartes f^k :

$$y_c(x) = \sum_{k} w_{kc}^{\chi} \sum_{ij} f_{ij}^k(x) \tag{1.25}$$

Pour respecter autant que possible les propriétés ci-dessus, il est proposé de trouver les poids w_{kc}^{x} qui minimisent la fonction de coût suivante :

$$Loss = \sum_{k} |y_c(x) - y_c^{\setminus k}(x) - w_{kc}^{\chi} \sum_{ij} f_{ij}^k(x)| + |y_c(x) - \sum_{k} w_{kc}^{\chi} \sum_{ij} f_{ij}^k(x)|, \tag{1.26}$$

où les premier et second termes entre valeurs absolues visent respectivement à respecter les

propriétés de sensibilité et de conservation. Les auteurs dérivent la solution analytique suivante :

$$w_{kc}^{x} = \sum_{ij} \frac{f_{ij}^{k}(x)}{\sum\limits_{i'j'} f_{i'j'}^{k}(x)} \times \frac{\partial y_c(x)}{\partial f_{ij}^{k}(x)} \tag{1.27}$$

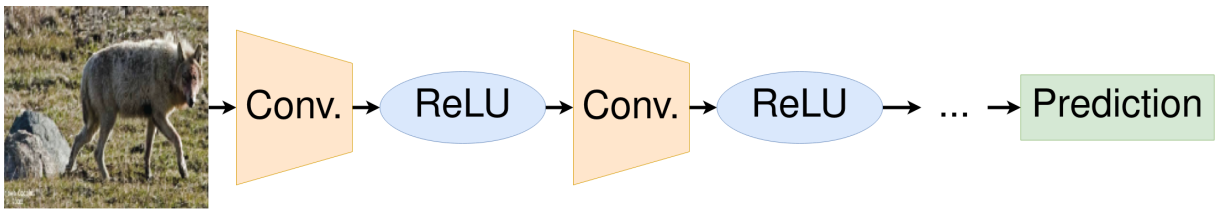
Approches par rétropropagation vers l'image d'entrée. Enfin, la dernière famille d'approche propose de rétropropager le score de la classe à expliquer vers l'image d'entrée pour obtenir une carte de saillance haute-résolution qui indique l'impact de chaque pixel sur la décision, i.e. $S \in \mathbb{R}^{H \times W}$. La méthode de base de ce type d'approche consiste à visualiser les gradients du score par rapport à l'image et on l'appelle simplement "carte de gradients" ("Gradient Map", GM), (aussi appelé *carte de sensibilité*) qui est équivalente à la méthode dite de "Déconvolution" dans le cas d'un CNN avec une activation ReLU :

$$\text{GM}_{ij}(x) = \frac{\partial y_c(x)}{\partial x_{ij}}, \tag{1.28}$$

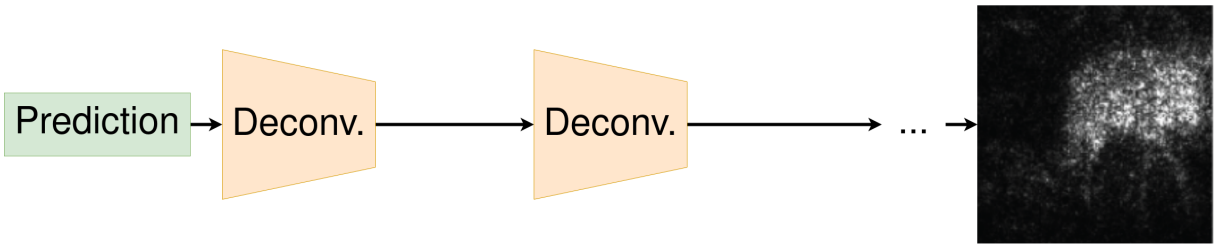
Springenberg et al. [136] proposent la *rétropropagation guidée* (“Guided Backpropagation”, GP) afin d'améliorer les cartes de gradient avec une idéee aussi utilisée par Grad-CAM++ : les gradients négatifs sont supprimés avec une activation ReLU, car ils correspondent à des neurones qui diminuent l'activité du neurone qui produit le score de la classe à expliquer, comme illustré en figure 1.8.

Également, Sundararajan et al. [142] proposent deux propriétés que les cartes de saillance devraient satisfaire, à savoir la *sensibilité* et l'*invariance à l'implémentation*. La sensibilité se définit comme suit. Si, lorsque l'on fait varier x_{ij} sans faire varier les autres pixels, $y_c(x)$ varie, alors une méthode avec la propriété de sensibilité doit faire varier la contribution de x_{ij} aussi. La deuxième propriété est respectée si une méthode n'est pas sensible à l'implémentation du modèle mais seulement à la fonction qu'il implémente. Par exemple si deux modèles implémentent la même fonction, i.e., s'ils produisent les mêmes prédictions avec les mêmes entrées, une méthode d'attribution invariante à l'implémentation doit produire les mêmes cartes de saillances pour ces deux modèles pour toute les entrées possibles. Les auteurs proposent une méthode qui respecte ces deux propriétés appelées "cartes de gradient intégrés" ("Integrated Gradients", IG) en agrégeant les cartes de gradients obtenues en interpolant entre l'image qu'on cherche à expliquer et une image de référence comme suit :

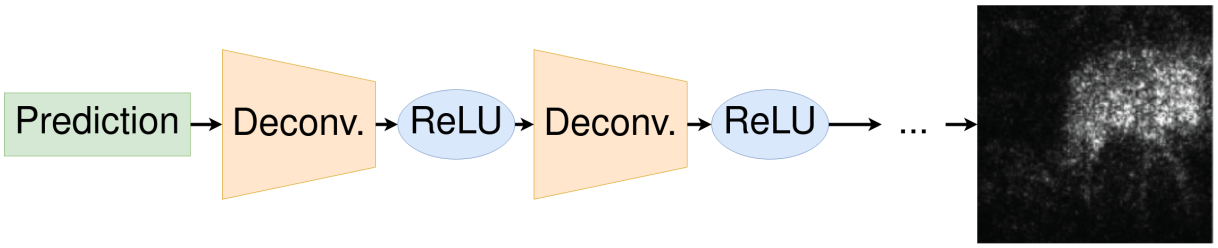
$$IG_{ij}(x) = (x_{ij} - \tilde{x}_{ij}) \int_0^1 \text{GM}_{ij}(\tilde{x} + \alpha(x - \tilde{x}))d\alpha, \tag{1.29}$$



(a) Inférence



(b) Déconvolution



(c) La méthode GP

FIGURE 1.8 – La méthode GP est équivalente à la méthode de déconvolution exceptée qu'elle remplace les gradients négatifs par des valeurs nulles pour ne visualiser que les pixels qui participent à l'activation de la classe à expliquer.

où $\tilde{x}$ est une image de référence. Comme évoqué précédemment, cette image doit représenter l'absence d'objet à classifier et les auteurs proposent donc d'utiliser une image noire à condition qu'avec une telle image, $y_c(\tilde{(x)}) \approx 0$. Une référence obtenue à partir d'un bruit gaussien pourrait aussi représenter cela mais les auteurs argumentent qu'un tel choix rajouterait du bruit dans la visualisation. Notez qu'en pratique cette intégrale est estimée avec une approximation Riemannienne, c'est-à-dire en calculant une moyenne de termes $\text{GM}_{ij}(\tilde{x} + \alpha(x - \tilde{x}))$ pour k valeurs de α suffisamment rapprochées. Plus tard, Smilkov et al. [135] argumentent que les GM varient de façon chaotique ce qui contribue au bruit observé sur ces cartes. En particulier, ils montrent qu'en interpolant à partir d'une image de référence à une image à expliquer, les dérivées partielles de y_c varient de façon abruptes et imprévisibles. Ce problème est bien connu dans la littérature des attaques antagonistes qui se basent sur cette propriété [165]. Pour pallier ce problème, ils proposent SmoothGrad pour améliorer IG en agrégeant les GM obtenues en perturbant l'image d'entrée avec un bruit gaussien :

$$\text{SmoothGrad}_{ij}(x) = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\sigma^2)}[\text{GM}_{ij}(x + \epsilon)] \tag{1.30}$$

En pratique, SmoothGrad est estimée avec des échantillons $\{\epsilon_k\}_k$ de la distribution normale :

$$\text{SmoothGrad}_{ij}(x) = \frac{1}{K} \sum_k \text{GM}_{ij}(x + \epsilon_k) \tag{1.31}$$

Ensuite, Adebayo et al. proposent une variante de SmoothGrad pour répondre au même problème, appelée VarGrad [2] qui est indépendante du gradient de la fonction de score [128]. Cette méthode utilise la variance des cartes de la méthode IG de l'entrée perturbée comme explication :

$$\text{VarGrad}_{ij}(x) = \mathcal{V}_{\epsilon \sim \mathcal{N}(0,\sigma^2)}[\text{GM}_{ij}(x + \epsilon)] \tag{1.32}$$

De façon similaire à SmoothGrad, on utilise des échantillons pour l'estimer :

$$\text{VarGrad}_{ij}(x) = \frac{1}{K} \sum_k (\text{GM}_{ij}(x + \epsilon_k) - \frac{1}{K} \sum_{k'} \text{GM}_{ij}(x + \epsilon_{k'}))^2 \tag{1.33}$$

$$= \frac{1}{K} \sum_k (\text{GM}_{ij}(x + \epsilon_k) - \text{SmoothGrad}_{ij}(x))^2 \tag{1.34}$$

On peut aussi citer deux autres méthodes appelées "apprentissage profond de caractéristiques importantes" ("Deep Learning of Important FeaTures", DeepLift) [130] et "propagation de pertinence par couche" ("Layerwise relevance propagation") [9]. Contrairement aux méthodes

évoquées précédemment dans ce paragraphe, DeepLIFT et LRP ne calculent pas de gradients. Cependant, ces méthodes se rapprochent néanmoins des méthodes basées sur les gradients car elles consistent à rétropropager des messages à partir de la sortie du modèle pour calculer la contribution de chaque pixel d'entrée et ainsi obtenir une carte d'explication à la résolution de l'entrée, à la manière de GP, IG, SmoothGrad, etc. Par exemple, DeepLIFT, qui a déjà été évoquée, postule l'équation suivante, dite de "somme vers différence" :

$$\sum_{k=1}^{K} \sum_{ij} C_{\Delta x_{ij} \Delta y} = y_c(x) - y_c(\tilde{x}), \tag{1.35}$$

Les auteurs de DeepLIFT démontrent l'existence de plusieurs règles de calcul permettant de calculer l'impact des activations $C_{\Delta x_{ij} \Delta y}$ des cartes sur la différence $y_c(x) - y_c(\tilde{x})$. Ces règles sont analogues à la rétropropagation des gradients dans le sens où sont d'abord calculées les contributions de chaque neurone de la couche précédent les scores de classification, puis les contributions de la couche qui la précède, etc., jusqu'à obtenir les contributions des pixels de l'image d'entrée. Selon les auteurs, l'intérêt de ne pas utiliser de gradient est de propager un signal d'importance même dans des situations où le gradient est nul et d'éviter les artefacts causés par des discontinuités de gradient.

Globalement on observe que, malgré la diversité de méthodes existantes, il reste un compromis entre la généricité des méthodes et leur efficacité. En effet, les méthodes basées sur la perturbation de l'image peuvent être appliquées à tout type de modèle mais sont coûteuses en calcul (RISE et LIME requièrent respectivement 8k et 15k inférences), et sont donc difficiles à appliquer sur des modèles avec un grand nombre de paramètre. Au contraire, les méthodes basées sur la pondération de cartes de caractéristiques ou la rétropropagation des gradients ne sont applicables qu'à des CNN, voir seulement aux CNN ne comportant qu'une couche dense comme CAM ou Grad-CAM++, mais ont un coût de calcul bien plus réduit.

Dans la suite, nous discutons de l'autre approche existante pour obtenir des cartes d'attention, les modèles d'attention spatiale.

1.2.2 Architectures d'attention

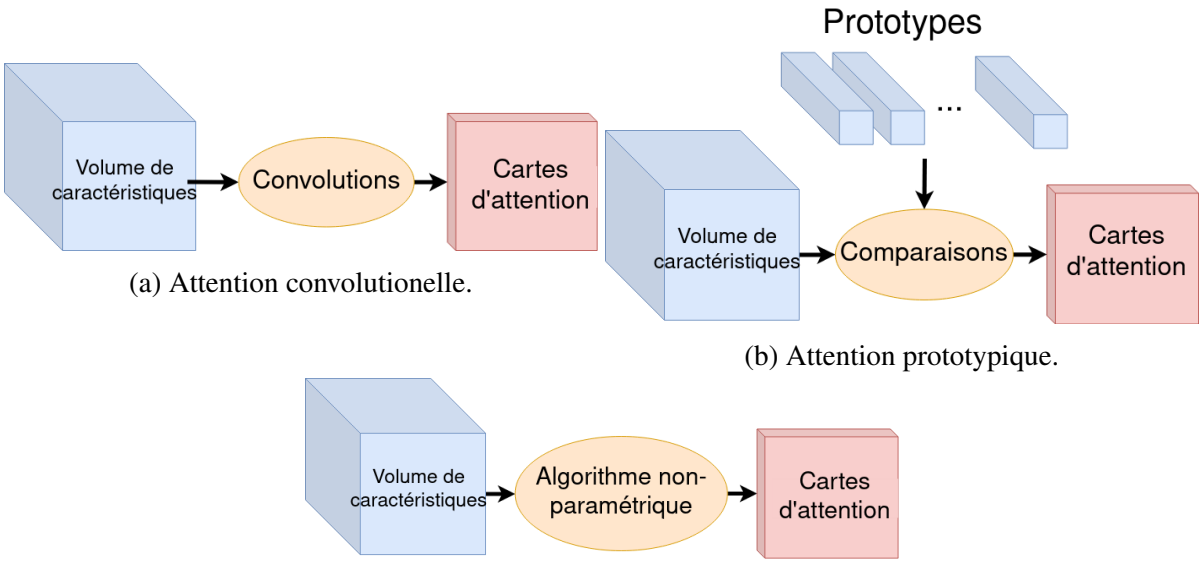
Les architectures d'attention ont récemment eu un gain d'intérêt suite à la proposition par Vaswani et al. d'une architecture majoritairement composés de couches d'attention, nommée le Transformer [148]. Ce modèle a montré son intérêt tout d'abord en traitement du langage naturel [148, 15] mais aussi plus récemment en vision par ordinateur [33, 18]. Cependant, ce type

d'architecture ne nous intéressera pas ici car elle ne répond pas à des besoins en interprétabilité. En effet les transformers génèrent plusieurs dizaines (voir centaines) de carte d'attention par couche et par inférence ce qui rend l'interprétation difficile.

De la même manière, les architectures d'attention convolutionelles comportant de multiples modules d'attention [154, 168, 28, 63] ne seront pas détaillées non plus car, comme les transformers, ils ne sont pas conçus pour l'interprétabilité mais pour la performance. En effet, au lieu d'un unique module d'attention situé avant l'agrégation spatiale comme les modèles évoqués ici, ces modèles intègrent de multiples couches d'attention entre les convolutions et génèrent donc un grand nombre de cartes d'attention. Cela pose plusieurs problèmes du point de vue de l'interprétabilité. Tout d'abord le grand nombre de cartes d'attention générées fait qu'il est difficile pour un utilisateur de toutes les visualiser. Ensuite, ces cartes n'étant pas suivies d'une agrégation spatiale, on peut argumenter qu'une zone qui est masquée par un module d'attention sera restaurée en étant mise en valeur par un autre module d'attention dans une couche suivante. Plus généralement, étant donné la complexité de ces modules, il est aussi difficile de comprendre comment ces cartes interagissent entre elles.

Pour ces raisons, nous nous intéresserons à des architectures qui ne comporte qu'un seul module d'attention. Dans la littérature ces architectures se divisent en trois catégories : les approches convolutionelles qui utilisent des couches de convolutions, les approches prototypiques qui comparent les vecteurs de caractéristiques à des prototypes appris et les approches non paramétriques qui utilisent des algorithmes non paramétriques. Ces trois approches sont illustrées en figure 1.9.

Approches convolutionelles. Les approches convolutionelles utilisent des couches de convolution pour calculer une carte d'attention. Soit $\mathbf{F} \in \mathbb{R}^{H' \times W' \times C}$ le tenseur contenant toute les cartes de caractéristiques de la dernière couche d'un CNN. Lors de l'inférence d'un CNN avec une couche d'agrégation spatiale par moyenne (comme les modèles ResNet par exemple [58]), les activations des cartes ont toutes le même poids dans le calcul du vecteur de caractéristique final. Cela peut être un problème car certaines activations correspondent à des zones de l'image qui ne contiennent pas des informations pertinentes pour la classification comme l'arrière-plan dans le cas de la classification d'images. Pour pallier ce problème, divers modules d'attention ont été proposés dans la littérature afin de permettre au modèle de se focaliser sur des parties spécifiques de l'image. Ces modules produisent des cartes d'attention qui donnent un poids différent aux différentes zones de l'image, permettant par exemple d'ignorer l'arrière-plan et de mettre en valeur l'objet d'intérêt. Ces cartes peuvent donc constituer des explications visuelles de la même



(a) Attention convolutionelle.

(b) Attention prototypique.

(c) Attention non paramétrique.

FIGURE 1.9 – Illustrations des trois types de module d'attention traités dans ce manuscrit.

manière que les cartes produites par les méthodes génériques évoquées précédemment, et seront donc aussi appelées *cartes de saillance*.

Le premier module d'attention étudié ici a été proposé par Hu et al. et est appelé "agrégation d'attention bilinéaire" ("Bilinear Attention Pooling", BAP) [64]. Ce module applique un couche de convolution 1×1 sur le tenseur $\mathbf{F}$ pour obtenir un carte d'attention $\mathbf{A} \in \mathbb{R}^{H' \times W' \times 1}$, comme illustré en figure 1.10. Ce module se combine donc avec un CNN qui produit des cartes de traits qui sont traités par le module BAP :

$$\mathbf{F} = \text{CNN}(x) \tag{1.36}$$

$$\mathbf{A} = \text{BAP}(\mathbf{F}) \tag{1.37}$$

Cette combinaison d'un CNN avec un module BAP est appelée dans ce manuscrit un CNN Bilinéaire ("Bilinear CNN", B-CNN). Ensuite $\mathbf{F}$ et $\mathbf{A}$ sont multipliés avec un produit externe comme suit :

$$\mathbf{F}_{att} = \mathbf{F} \times \mathbf{A}, \tag{1.38}$$

où $\mathbf{F}_{att} \in \mathbb{R}^{H' \times W' \times C}$. Enfin, une agrégation par moyenne est appliquée sur les dimensions spatiales pour obtenir le vecteur de caractéristiques final $f \in \mathbb{R}^C$ qui sera passé à une couche linéaire afin d'obtenir une prédiction. Les activations du tenseur $\mathbf{F}_{att}$ ont été amplifiées ou

réduites en fonction du poids indiqué par la carte $\mathbf{A}$, ce qui permet au modèle de donner plus d'importance à l'objet et d'ignorer l'arrière-plan. Cependant, si l'objet est composé de plusieurs parties importantes comme dans le cas de la classification fine, la carte indiquera chacune de ces parties comme importante. Le vecteur de caractéristique obtenu sera alors un agrégat de l'information contenu par les différentes parties. Afin de maximiser l'expressivité du modèle, l'approche dominante en classification fine consiste à extraire un vecteur de caractéristique par partie importante de l'objet et à concaténer les vecteurs obtenus avant de passer le vecteur final à la couche de classification. En pratique, la convolution 1×1 produit $N > 1$ cartes d'attention, c'est-à-dire un tenseur de taille $H' \times W' \times N$ où N est considéré comme le nombre de parties maximum de l'objet. Suite au produit externe le tenseur obtenu a pour dimension où $H' \times W' \times N \times C$ et après agrégation spatiale on obtient une matrice de caractéristiques F de dimension $N \times C$ qui est aplatie en un vecteur $f \in \mathbb{R}^{NC}$ qui est lui-même passé à la couche de classification linéaire. Ce mécanisme permet au modèle de concentrer son attention sur plusieurs endroits différents sans perdre d'information lors de l'agrégation spatiale. Le module BAP a été réutilisé par la suite par d'autres auteurs notamment dans le cadre de transfert d'apprentissage [71], de stratégies d'augmentation de données [22] et d'apprentissage causal [119].

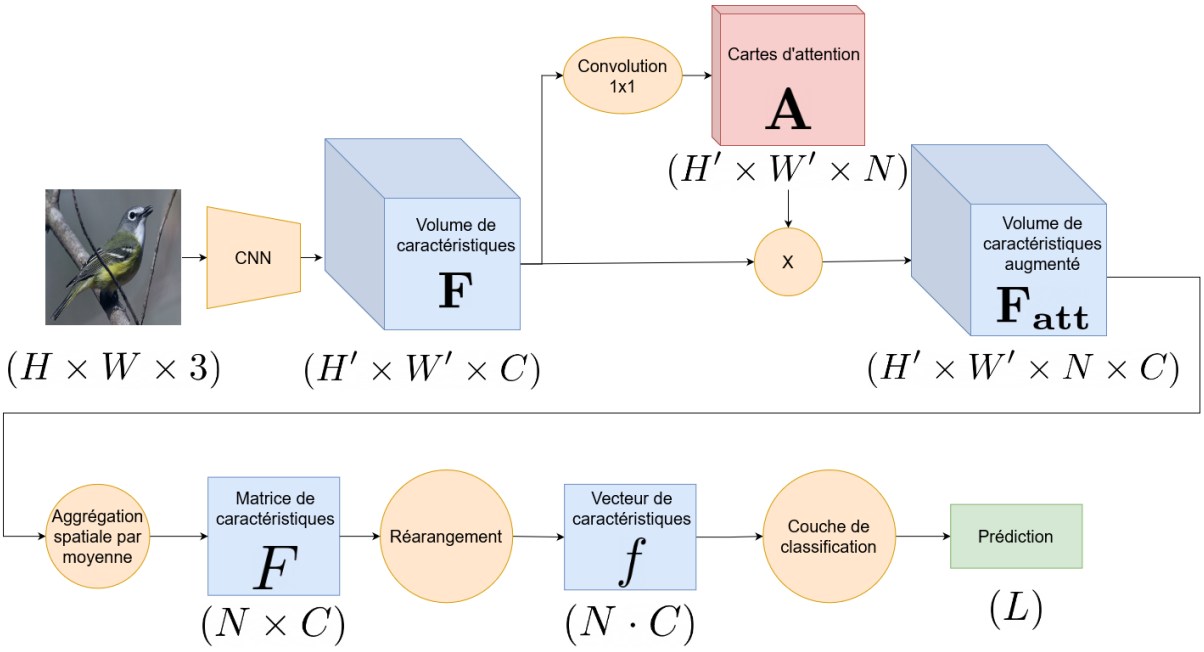


FIGURE 1.10 – Illustration d'un B-CNN. Une convolution 1×1 permet de générer les cartes d'attention. Celles-ci sont ensuite multipliées par les cartes de caractéristiques avant d'appliquer une agrégation spatiale. Enfin, la matrice obtenue est réarrangée en un vecteur qui est passé à la couche de classification.

Avec un modèle nommé CNN multi-attention ("Multi-Attention CNN", MA-CNN), Zheng et al. proposent d'obtenir une carte d'attention en calculant une moyenne pondérée de cartes de caractéristiques où les poids sont produits par une couche dense [172]. Plus précisément, les auteurs entraînent N couches denses à produire des poids $\{w^n\}_n$ pour générer N cartes d'attention à partir des cartes de caractéristiques $\mathbf{F}$ comme suit :

$$w^1, w^2, ..., w^N = d^1(\mathbf{F}), d^2(\mathbf{F}), ..., d^N(\mathbf{F}) A^n \quad = \sum_k w_k^n f^k, \mathbf{A} = concat(A^1, ..., A^N) \quad (1.39)$$

où $w^n \in \mathbb{R}^K$, $A^n \in \mathbb{R}^{H' \times W'}$, $\mathbf{A} \in \mathbb{R}^{H' \times W' \times N}$ et $concat$ est un opérateur de concaténation. Les auteurs proposent de régulariser les modules d'attention d^n en les entraînant à donner un poids similaire aux cartes de caractéristiques f^k qui ont tendance à s'activer aux mêmes endroits de l'image et avec un terme de la fonction de coût qui incite chaque carte f^k à n'avoir un poids important que dans le calcul d'une carte d'attention. On note ici que l'attention de MA-CNN est bien spatiale, car elle donne un poids à chaque position spatiale ($A^n \in \mathbb{R}^{H' \times W'}$), quand bien même les poids w^n s'appliquent sur chaque carte f^k et donc sur la dimension des canaux.

Des mécanismes d'attention avec plusieurs couches de convolution ont aussi été proposés. Par exemple, Fukui et al. ont proposé un modèle appelé Réseau avec branche d'attention ("Attention Branch Network", ABN) qui génère une carte d'attention $\mathbf{A} \in \mathbb{R}^{H' \times W' \times 1}$ avec une suite de convolution entrelacée avec des couches de normalisation de lot et des activations ReLU terminée par une activation sigmoïdale [43]. On ajoute ensuite un tenseur de 1 à la carte d'attention avant de l'appliquer sur les cartes de traits comme suit :

$$\mathbf{F}_{att} = \mathbf{F} \times (\mathbf{A} + 1) \quad (1.40)$$

Les auteurs justifient le choix d'ajouter un tenseur unitaire en argumentant que cela permet au module d'attention de mettre en valeur des parties importantes de l'image en empêchant les parties moins importantes d'être ignorées. En effet, les valeurs de $\mathbf{A}$ sont comprises entre 0 et 1 suite à l'application de la fonction sigmoïde. Le ratio maximal entre la zone la plus importante et la zone la moins importante peut donc tendre vers l'infini : $\mathbf{A}_{max}/\mathbf{A}_{min} \to +\infty$ et ajouter un tenseur unitaire permet d'imposer une borne supérieure à ce ratio : $\sup_{\mathbf{A}}(\mathbf{A}_{max}+1)/(\mathbf{A}_{min}+1) = 2$. Ce type d'attention est appelé attention résiduelle [34, 43, 154, 168] car on peut la formuler ainsi :

$$\mathbf{F}_{att} = \mathbf{F} + \mathbf{F} \times \mathbf{A}, \quad (1.41)$$

où le second terme est vu comme le résidu, à la manière d'un bloc résiduel de ResNet [58].

Au lieu de traiter les images une par une, Zhuang et al. [34] proposent un mécanisme d'attention croisée où les images sont traitées par pairs et où les informations extraites avec une image comme attention sur les canaux sur l'autre image. Soit x_1 et x_2 deux images et f_1 et f_2 les vecteurs de caractéristiques correspondants extraits avec un CNN. Un vecteur mutuel est d'abord calculé :

$$f_m = MLP(concat(f_1, f_2)), \tag{1.42}$$

où MLP désigne un perceptron multi-couches ("Multi Layer Perceptron", MLP). Ensuite, un vecteur de poids sur les canaux est obtenu en utilisant f_m pour guider l'attention :

$$g_i = sigmoid(f_m \cdot f_i), \forall i \in \{1, 2\}, \tag{1.43}$$

où $\cdot$ désigne le produit élément par élément. Les vecteurs de caractéristiques finaux sont obtenus en utilisant les vecteurs g_1 et g_2 comme attention résiduelle :

$$f_1^{self} = f_1 + g_1 \cdot f_1 \tag{1.44}$$
$$f_1^{other} = f_1 + g_2 \cdot f_1 \tag{1.45}$$
$$f_2^{self} = f_2 + g_2 \cdot f_2 \tag{1.46}$$
$$f_2^{other} = f_2 + g_1 \cdot f_2, \tag{1.47}$$

where the superscripts $self$ and $other$ respectively identify the vectors whose channels are masked with the attention g_i calculated from itself and the other vector.

Approches prototypiques. Au lieu d'utiliser des couches convolutionelles, Chen et al. proposent un modèle appelé ProtoPNet, qui utilise des prototypes représentés par des vecteurs de paramètres appris pendant l'entraînement. Durant l'inférence, on calcule la similarité entre ces prototypes et l'information extraite par le CNN, comme illustré en figure 1.11.

Plus précisément, une carte d'attention $\mathbf{A}^n$ est calculée en comparant un prototype p^n à chaque vecteur de caractéristique $\mathbf{F}_{ij}$:

$$\mathbf{A}_{ij}^n = \log\left(\frac{||\mathbf{F}_{ij} - p^n||_2^2 + 1}{||\mathbf{F}_{ij} - p^n||^2 + \epsilon}\right) \tag{1.48}$$

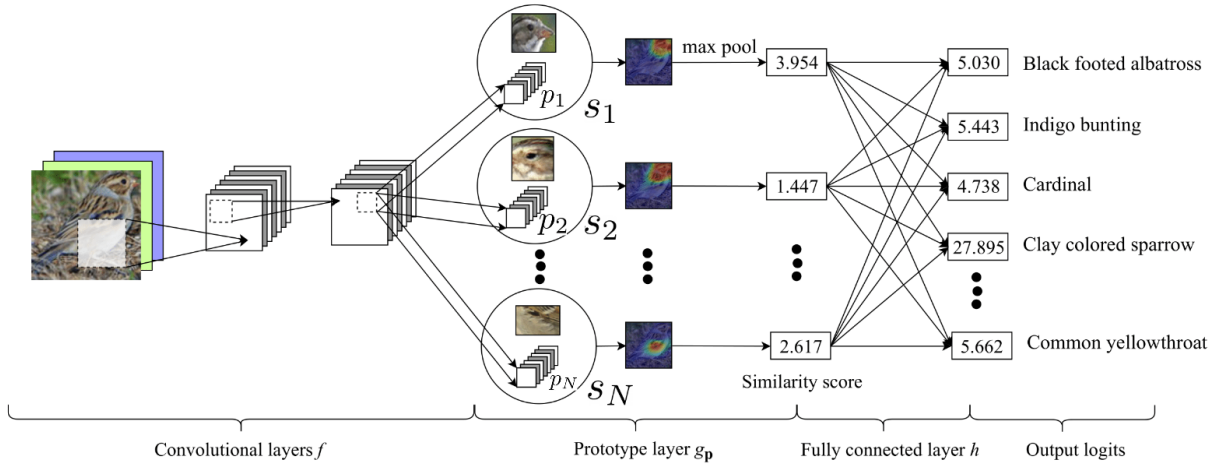


FIGURE 1.11 – Illustration de l'architecture prototypique de ProtoPNet. Extrait de [21]. Les vecteurs de caractéristiques extraits par les couches convolutionelles sont comparées aux prototypes, ce qui produit des cartes d'attention et un vecteur de similarité. Ce vecteur est ensuite passé à la couche de classification pour produire une prédiction.

Ensuite, un score global de similarité s^k est calculé avec une agrégation par maximum :

$$s_n = \underset{ij}{max} \mathbf{A}^n_{ij} \tag{1.49}$$

La similarité s_n obtenue représente à quel point le prototype p^n existe dans l'image. La couche de classification linéaire y prend ensuite en entrée le vecteur de similarité s pour produire une prédiction. Chen et al. proposent également d'assigner chaque prototype à une classe, chaque classe disposant de $N//L$ prototypes, où L est le nombre de classes. On note y_{gt} le label de l'image x et $\mathcal{P}_y$ l'ensemble des prototypes de la classe y. L'entraînement se déroule ensuite en trois phases. Premièrement, les paramètres du CNN qui sont optimisés pour que les vecteurs de caractéristiques extraits soit proches des prototype de la bonne classe et et en même temps, les prototypes sont optimisés pour qu'on trouve au moins un prototype dans chaque image avec la fonction de coût suivante :

$$Loss = CE(y(x), y_{gt}) + \lambda_1 C + \lambda_2 S, \tag{1.50}$$

où CE est la fonction d'entropie croisée et les termes C et S ont respectivement pour but de rapprocher les vecteurs de caractéristiques d'un prototype de la bonne classe et de les éloigner

des prototypes des autres classes. Ces termes sont définis comme suit :

$$C = \min_{p^u \in \mathcal{P}_y} \min_{\mathbf{F}_{ij}} ||p^u - \mathbf{F}_{ij}||_2^2 \tag{1.51}$$

$$S = -\min_{p^u \notin \mathcal{P}_y} \min_{\mathbf{F}_{ij}} ||p^u - \mathbf{F}_{ij}||_2^2 \tag{1.52}$$

Durant cette première phase, les paramètres de h sont fixés à des valeurs constantes. Soit w_{nc} le poids du prototype p_n sur la classe c. Alors, si p_n est un prototype de la classe c alors w_{nc} est fixé à 1 et sinon il est fixé à -0.5. Cela a pour effet de forcer le réseau à apprendre des prototypes qui sont effectivement représentatif de la classe. Durant la deuxième phase d'entraînement, les prototypes sont remplacés par les vecteurs de caractéristiques qui leur sont le plus proches afin de pouvoir interpréter chaque prototype comme un patch d'une image. En effet, chaque vecteur de caractéristique correspond à un patch de taille $H/H' \times W/W'$ sur l'image d'entrée. Soit p_n un prototype de la classe c. Ce vecteur est remplacé comme suit : $p_n = \underset{\mathbf{F}_{ij}, x \in X_c}{argmin} ||\mathbf{F}_{ij} - p_n||_2$, où X_c est l'ensemble des images de la classe c. On parcourt donc tous les vecteurs de caractéristiques extraits parmi toutes les images de la classe c afin de trouver un vecteur qui soit le plus proche de p possible afin de ne pas altérer significativement le comportement global du modèle. La troisième et dernière étape de l'entraînement consiste à optimiser seulement les paramètres w_nc de la couche de classification h. Les auteurs utilisent pour cela l'entropie croisée auquel ils ajoutent un terme de parcimonie sur les poids w_{nc} :

$$Loss = CE(y(x), y_{gt}) + \lambda_3 \sum_{nc} |w_{nc}| \tag{1.53}$$

Chen et al. argumentent que ce terme de régularisation défavorise les modèles qui raisonnent par la négative, c'est-à-dire les modèles avec des poids w_{nc} négatifs. On remarque cependant qu'avec ce terme de régularisation, un poids négatif pénalisera le modèle autant qu'un poids positif de même valeur absolue et il serait peut-être plus efficace de ne prendre en compte que les poids négatifs.

Récemment, un modèle appelé TesNet a été proposé par Wang et al [157]. Afin d'améliorer l'interprétabilité de l'espace des prototypes, Wang et al. proposent de forcer les $N//L$ prototypes d'une même classe à former un sous-espace de dimension $N//L$ avec un terme de fonction de coût qui favorise l'orthogonalité :

$$Loss_{ortho} = \sum_c ||P^c P^{c\top} - \mathbb{I}||_F^2, \tag{1.54}$$

où P^c est une matrice formée par les prototypes assignés à la classe c et $\mathbb{I}$ est la matrice identité de taille $N//L$. Les auteurs utilisent également un terme basé sur la métrique de projection [56] pour séparer les sous-espaces les uns des autres :

$$Loss_{ss} = \frac{1}{\sqrt{2}} \sum_{c_1} \sum_{c_2} ||{P^{c_1}}^\top P^{c_1} - {P^{c_2}}^\top P^{c_2}||_F \tag{1.55}$$

Enfin, les termes C et S de l'équation (1.50) sont modifiés afin d'utiliser la similarité cosinus au lieu de la distance euclidienne pour mesurer la compatibilité entre les vecteurs et les prototypes. Les auteurs expliquent ce choix en argumentant que la distance euclidienne revient à faire l'hypothèse que les prototypes sont distribués de façon gaussienne, et que cela n'est pas approprié pour des structures de données complexes. Notez cependant que les auteurs ne démontrent pas comment la distance L_2 implique une distribution gaussienne ni en quoi ce n'est pas approprié pour des données complexes. Aussi, les auteurs montrent expérimentalement l'intérêt de TesNet par rapport à ProtoPNet en termes de précision mais pas en termes d'interprétabilité.

En se basant aussi sur le modèle ProtoPNet, Nauta et al. proposent ProtoTree, un modèle où les prototypes sont arrangé en un arbre binaire et l'image est dirigée à gauche ou à droite d'un noeud en fonction de sa similarité avec le prototype associé au noeud [107]. Afin de faciliter l'entraînement, les images sont dirigées de façon continue dans l'arbre et à chaque nouveau noeud. On peut donc calculer la probabilité que l'image arrive à chaque feuille s^l comme suit :

$$s^l = \prod_{n \in \mathcal{N}_l} s_n, \tag{1.56}$$

où $\mathcal{N}_l$ désigne l'ensemble des noeuds sur le chemin pour atteindre la feuille s^l. Comme dans un arbre de décision, on associe également à chaque feuille une distribution $\sigma_l \in \mathbb{R}^L$ sur les classes qui sont définies par des paramètres entraînables du modèle. La prédiction finale est obtenue en agrégeant les distributions avec les probabilités d'arriver à chaque feuille :

$$y(x) = \sum_{n \in \mathcal{N}_l} s_n \times \sigma_l \tag{1.57}$$

Les auteurs montrent que cette architecture peut être rendue plus interprétable en élaguant des branches ou en la rendant déterministe.

Au lieu d'assigner chaque prototype à une seule classe, Rymarczyk et al. ont proposé un modèle nommé ProtoPool où un prototype peut être utilisé par plusieurs classes [123]. Plus précisément, on définit N *emplacements* avec des distributions sur les prototypes associées

$q^n \in \mathbb{R}^N$ où $n \in [1, ..., N]$, comme illustré en figure 1.12.

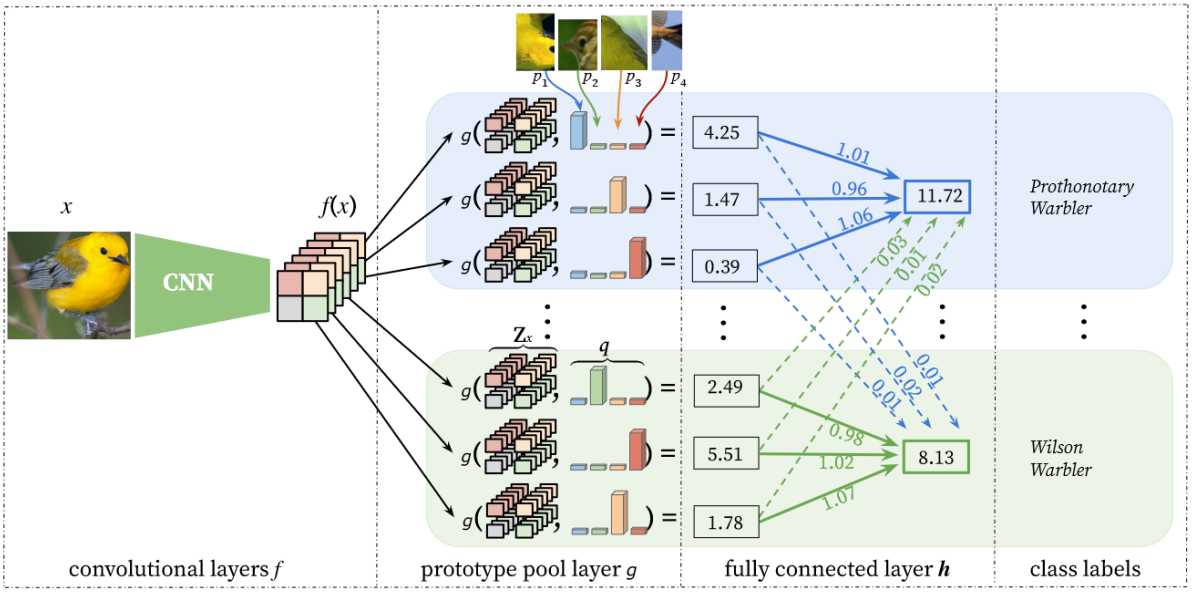


FIGURE 1.12 – Illustration du modèle ProtoPool. Afin de partager les prototypes entre les classes, les prototypes sont assignés de façon continue à des emplacements, où un prototype peut appartenir à plusieurs emplacements. Extrait de [123]

Avec ProtoPool, Rymarczyk et al. proposent que ce ne soit plus les prototypes qui soient assignés à une classe à l'avance mais les emplacements. Pour obtenir les cartes d'attention A^n, les vecteurs de caractéristiques sont cette fois comparés à une moyenne pondérées des prototypes, où les poids sont donnés par les q^n. En pratique, les q^n sont des paramètres entraînables ce qui permet au modèle de partager un prototype entre plusieurs classes. Les auteurs proposent aussi de modifier l'équation (1.49) en argumentant qu'elle a plusieurs problèmes. Tout d'abord, étant donné que les cartes d'attention A^n sont agrégées spatialement par maximum, il est possible d'obtenir une similarité s_n élevée même si tous les éléments de A^n ont une valeur élevée, i.e. tous les vecteurs de caractéristiques sont similaires au prototype p^n. Selon Rymarczyk et al., cela est un problème car les prototypes peuvent donc être similaires à l'arrière-plan de l'image et malgré tout produire une grande similarité s^n. Le second problème est que pendant l'entraînement, l'agrégation par maximum signifie que le gradient ne passe que par la partie la plus active de la carte A^n ce qui rendrait l'optimisation plus difficile. Afin de pallier ces problèmes, la fonction d'agrégation suivante est proposée afin de forcer le modèle à se concentrer sur des

parties saillantes mais réduites de l'image :

$$s_n = \underset{ij}{max}\mathbf{A}^n_{ij} - \frac{1}{H'W'}\sum_{ij} A^n_{ij} \tag{1.58}$$

Le fait de soustraire la moyenne au maximum permet d'ignorer les cas où un prototype est similaire à toute l'image. Les auteurs montrent que ce choix produit des cartes d'attention concentrées sur une partie réduite de l'image et conduit les utilisateurs à qualifier les prototypes comme plus "distinctif" de la classe.

Xiao et al. argumentent que les modèles ProtoPNet et ProtoTree apprennent des prototypes qui sont n'ont pas de sens ou sont non pertinents et qui fournissent peu d'explication pour la prédiction [162]. Par exemple, la figure 1.13 montrent que ProtoTree et ProtoPShare (l'ancien nom de ProtoPool) détectent un prototype au bord de l'assiette. Les auteurs argumentent que ces prototypes peuvent aider à classer cette image comme "salade grecque", mais ils n'aident pas à interpréter cette décision. Pour pallier ce problème, ils proposent donc deux modifications à ProtoPNet. Premièrement, afin de partager les prototypes entre les classes pour ne pas perdre en capacités de généralisation, le terme S de l'équation (1.50) est supprimé ce qui permet aux mots visuels appris d'être partagés entre les catégories. Cette idée ayant déjà été proposée par ProtoPool, ils proposent également d'inciter les cartes d'attention correspondants aux prototypes les plus présents à être proches de la carte de saillance produite par Grad-CAM avec le terme de fonction de coût suivant :

$$Loss_a = ||Grad - CAM(x) - \frac{1}{N}\sum_{n \in top-K(s^n)} A^n||^2 \tag{1.59}$$

Les motivations de Xiao et al. soulèvent cependant un problème. En effet, ils argumentent qu'un prototype qui représente le bord d'une assiette n'est pas pertinent dans le cadre de la tâche de reconnaissance de plat. On peut critiquer cet argument en disant que le but d'un modèle interprétable n'est pas d'utiliser les mêmes indices visuels qu'un humain pour exécuter une tâche mais d'être capable d'expliquer quels sont les indices utilisés. L'intérêt d'un modèle interprétable est qu'il permet justement de s'apercevoir du fait que le modèle utilise de mauvais indices visuels comme le bord de l'assiette avec ProtoTree et ProtoPool.

Nauta et al. [108] ont par la suite proposé de générer des explications aux similarités s^n pour aider l'utilisateur à comprendre pourquoi le modèle estime qu'un prototype est similaire à un certain patch de l'image. Pour cela, ils constituent des versions alternatives des images d'entraînement en modifiant certains aspects de l'image comme le contraste, la couleur ou la

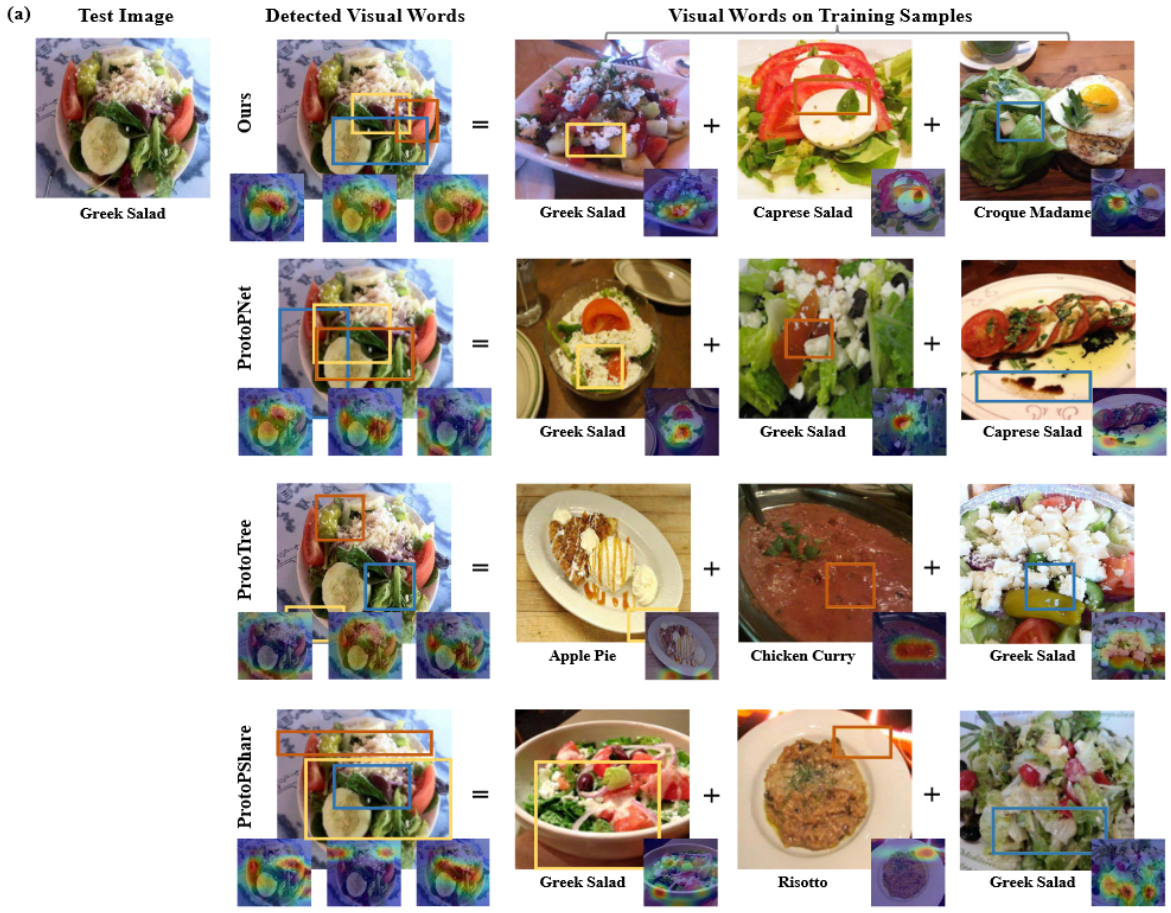


FIGURE 1.13 – Exemples de prototypes identifiés par plusieurs modèles prototypiques. Les auteurs de [162] questionnent la qualité de certains prototypes extraits par ProtoTree et ProtoPool (i.e. ProtoShare) car ils représentent des indices visuels non pertinent pour la tâche comme le bord de l'assiette (colonnes 2 et 3 respectivement) dans le cadre de la reconnaissance de plat.

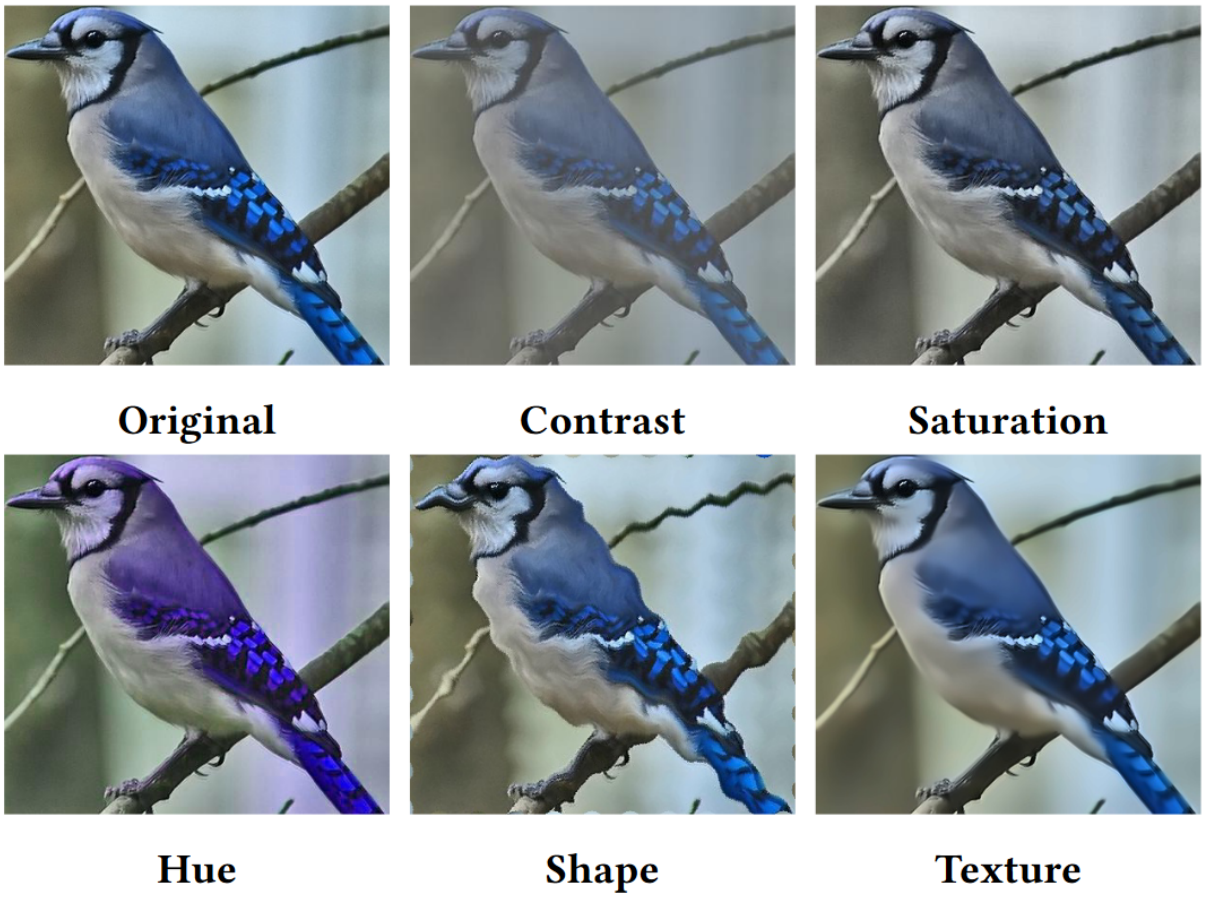


FIGURE 1.14 – Illustration des transformations appliquées sur les images par Nautal et al. [108] pour déterminer l'importance de chaque aspect (contraste, couleur, forme, et.) dans le score de similarité obtenus par le modèle. Extrait de [108].

texture, illustrées en figure 1.14. Ensuite, les auteurs comparent les similarités obtenues avec ces versions alternatives avec les similarités obtenues avec l'image originale. En fonction de la variation (positive ou négative) qu'a créée chaque type de modification de l'image, on en déduit quels sont les aspects de l'image originale qui ont été les plus importants pour déterminer qu'un patch est similaire à un prototype.

Afin d'enrichir les explications de ProtoPNet, il a aussi été proposé d'utiliser des prototypes déformables par Donnelly et a. [32]. Pour cela les prototypes ne sont pas représentés par un vecteur, qui peut être vu comme un tenseur de taille $1 \times 1 \times D$ mais par plusieurs, sous la forme d'un tenseur de taille $H'' \times W'' \times D$ où $H'' = W'' = 2$ ou 3. Ensuite, chaque vecteur qui compose le prototype est interprété comme une partie prototypique et lors de l'application du prototype sur l'image, un module dédié applique une translation sur chaque partie afin de

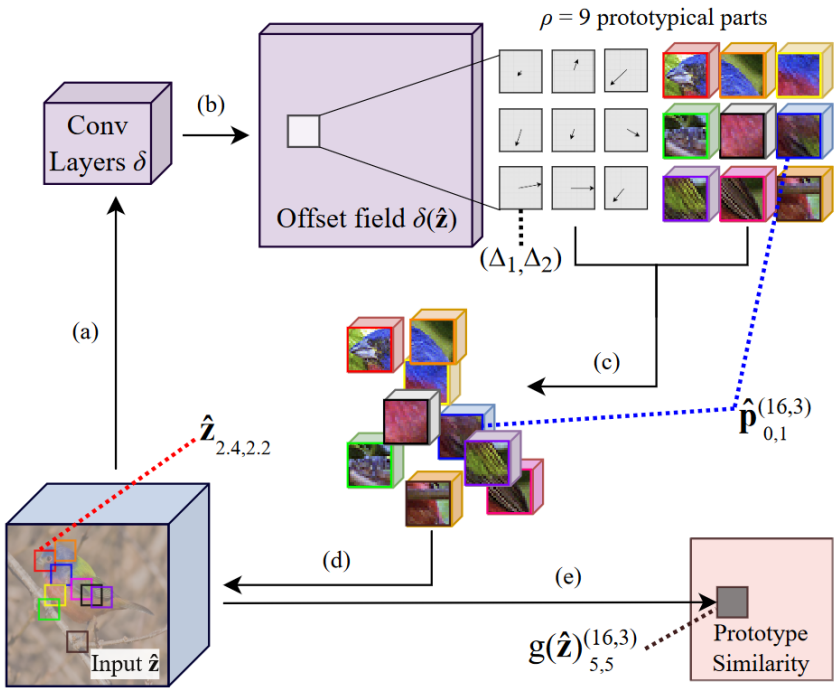


FIGURE 1.15 – Illustration du modèle ProtoPNet déformable. Une fois les cartes de traits calculées (a), elles sont passées à un module qui prédit des translations des parties des prototypes (b). Cela permet d'appliquer les prototypes en tenant compte de la pose de l'objet (c et d) et d'obtenir des scores de meilleurs scores de similarité (e) ce qui améliore la précision du modèle. Extrait de [32].

s'adapter à la pose de l'objet, comme illustré en figure 1.15.

Également basés sur le travail de Chen et al., Huang et al. [67] proposent non pas de calculer des similarités mais des probabilités de présence et pénaliser les probabilités éloignées de 0 ou 1. La conséquence est que le modèle est encouragé à apprendre des prototypes dont il est certain de l'absence/présence à chaque image. Les auteurs argumentent que cela facilite l'apprentissage de la reconnaissance des parties de l'objet, ce qui génère des cartes d'attention plus précise et donc améliore l'interprétabilité. Dans la suite de ce manuscrit, ce modèle est appelé "Interpretability By Parts" (IBP) "interprétabilité par partie".

Le modèle ProtoPNet a aussi inspiré des variantes plus incrémentales qui proposent par exemple de laisser le modèle raisonner de façon négative [134] ou d'employer des prototypes définis par non pas un mais plusieurs vecteurs [133].

Approches non paramétriques. Nous évoquons ici les deux seules approches que nous avons trouvé dans la littérature qui ne mettent pas en jeu de paramètres entraînables dans leur module

d'attention.

Tout d'abord, il est possible de simplement calculer la moyenne des cartes de caractéristiques pour obtenir une carte d'attention qui permet de localiser correctement les objets d'intérêt [23], comme l'ont montré Choe et al. Ensuite, l'autre module d'attention non paramétrique a été proposé par Zheng et al. et est basée sur l'agrégation de cartes de caractéristiques similaires [173]. Tout d'abord les cartes f^k sont normalisées spatialement avec une fonction d'activation softmax :

$$\mathbf{F}_{norm} = softmax_{spat}(\mathbf{F}) \tag{1.60}$$

On note f^k_{norm} les cartes après la normalisation. On a donc :

$$\sum_{ij} f^k_{norm\ ij} = 1 \tag{1.61}$$

Les cartes sont ensuite applaties pour n'avoir qu'une seule dimension spatiale $\mathbf{F}_{flat-norm} = flat(\mathbf{F}_{norm}) \in \mathbb{R}^{H'W' \times K}$. On calcule ensuite une matrice de similarité $M^{sim} \in \mathbb{R}^{K \times K}$ entre les cartes de caractéristiques Le but de cette matrice est d'indiquer pour toute paire de cartes de caractéristiques à quelle point elle s'active aux mêmes endroits de l'image et elle est calculée comme suit :

$$M^{sim} = softmax(\mathbf{F}^T_{flat-norm} \times \mathbf{F}_{flat}), \tag{1.62}$$

où $\times$ désigne le produit matriciel et où la normalisation $softmax$ s'applique sur la seconde dimension de M^{sim}. On obtient ainsi les cartes d'attention A^k dont les activations sont calculées comme suit :

$$A^k_{ij} = \sum_{k'} M^{sim}_{kk'} f^{k'}_{ij} \tag{1.63}$$

1.2.3 Synthèse

Des exemples de cartes de saillance générées par les algorithmes mentionnés ici sont visibles en figure 1.16.

Modèles d'attention et méthodes post-hoc. L'une des principales différences entre les méthodes post-hoc et les modèles d'attention est le coût de calcul associé à chaque approche. Si l'on considère uniquement le coût d'apprentissage, les méthodes post-hoc sont plus efficaces car elles ne nécessitent pas de ré-entraîner un modèle. Au contraire, si l'on considère le coût d'inférence, les modèles d'attention peuvent être plus efficaces, notamment par rapport aux méthodes post-hoc basées sur des perturbations. En effet, les méthodes post-hoc peuvent né-

cessiter un nombre d'échantillons allant de quelques milliers à dizaines de milliers, alors que les modèles d'attention ne nécessitent qu'une seule inférence pour obtenir une carte d'attention. Les développeurs souhaitant générer des cartes de saillance devraient alors être guidés par un compromis entre le coût de l'entraînement et le coût de l'inférence.

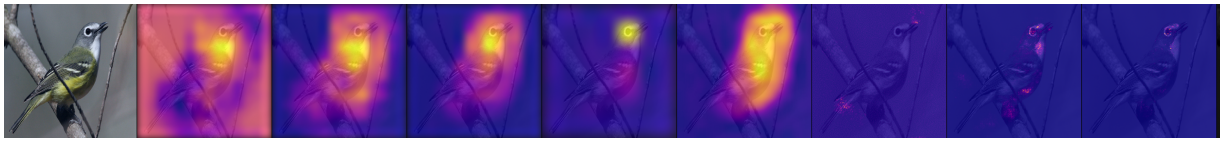
Cependant, le critère le plus important pour comparer ces deux types d'approches est leur interprétabilité, tant en termes de fiabilité que de compréhensibilité par les utilisateurs. Malgré cela, il n'existe actuellement pas de consensus sur la méthodologie à employer pour évaluer l'interprétabilité et il n'existe que quelques travaux qui comparent ces approches (cf. le chapitre 6 de ce manuscrit et le travail de Bastings et al.[10]).

Actuellement, on ne peut donc pas distinguer les méthodes post-hoc des modèles d'attention du point de vue de leur interprétabilité. De plus, cette thèse se concentre en grande partie sur l'évaluation de l'interprétabilité et peu sur l'amélioration des performances de classifications des modèles. Le coût en inférence est ici donc plus important que le coût d'entraînement. Par conséquent, nous choisissons de présenter dans le chapitre 3 de ce manuscrit une nouvelle approche de génération de cartes de saillance qui s'inscrit dans la famille des modèles d'attention.

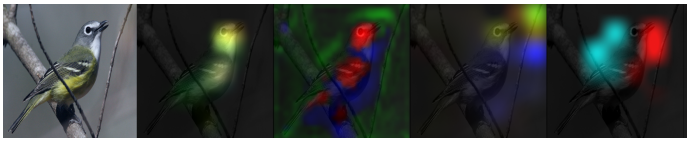
Attention convolutionelle, prototypique et non-paramétrique. La plupart des modèles d'attention proposés sont des approches prototypiques ou des approches convolutionelles. De plus, ces deux types d'approches sont similaires par la manière dont sont calculées les cartes d'attention $\mathbf{A}$. Dans les deux approches, elles permettent d'agréger l'information de façon spatiale en donnant plus de poids aux zones jugées pertinentes. Dans le cas des approches convolutionelles, $\mathbf{A}$ est utilisée pour donner un poids à chaque position spatiale et procéder à une agrégation par moyenne pondérée. Les modèles prototypiques proposent une approche similaire excepté qu'il s'agit d'une agrégation par maximum. On remarque aussi que dans le cas des approches convolutionelles, la couche d'attention est un bloc de convolution qui consiste aussi à calculer l'interaction entre des paramètres (les noyaux) appris pendant l'entraînement et les vecteurs de caractéristiques. Quand ce bloc n'est constitué que d'une seule convolution [93], les noyaux peuvent être interprété comme des prototypes, à la manière des approches prototypiques. Enfin, parmi les modèles cités ici, seules les approches prototypiques sont pensées pour être interprétable. Afin de proposer une approche d'attention interprétable originale, nous proposerons dans le chapitre 3 le premier modèle d'attention interprétable non paramétrique à notre connaissance.

Dans la section suivante, nous discutons des méthodes proposées dans la littérature pour évaluer la qualité des cartes de saillance produites par les méthodes post-hoc et les modèles d'attention. Nous nous attardons particulièrement sur les métriques de fiabilité qui évaluent la

fidélité d'une explication à la prédiction du modèle.



(a) Méthodes post-hoc. De gauche à droite : Grad-CAM, Grad-CAM++, AM, RISE, Score-CAM, Guided Grad-CAM, VarGrad et SmoothGrad.



(b) Modèles d'attention. De gauche à droite : B-CNN, IBP, ProtoPNet et ProtoTree.

FIGURE 1.16 – Exemples de carte de saillance générées par diverses méthodes d'explications et modèles d'attention.

1.3 Évaluation des cartes de saillance

Face au nombre croissant de méthodes d'explications, Adebayo et al. ont proposé une évaluation des approches par rétropropagation vers l'image d'entrée permettant de déterminer quel type d'explication ces méthodes peuvent ou ne peuvent pas produire [3]. Concrètement, ils mesurent la dépendance des cartes produites par diverses méthodes aux paramètres des modèles expliqués. Ils montrent par exemple que plusieurs méthodes produisent des cartes qui dépendent peu des paramètres des dernières couches du modèle ou de la classe de l'image, notamment GP, questionnant l'efficacité de ces méthodes dans ce cadre d'évaluation. Au contraire, des travaux récents ont montré que GP permet de détecter de façon efficace des anomalies dans les images d'entrée [29].

D'autres travaux ont aussi interrogé la fidélité des méthodes d'explications [16] ou des modèles d'attention [74, 160, 10].

Évaluer l'interprétabilité d'un modèle interprétable ou d'une méthode d'explication est difficile. Comme évoqué précédemment, il n'y a pas encore de définition acceptée globalement ni de protocole standard d'évaluation ce qui conduit les auteurs a proposer plusieurs stratégies pour pallier ce problème.

La méthode utilisée par tous les travaux cités ici consiste à analyser de façon qualitative les visualisations de cartes de saillance produites par les méthodes et modèles [127, 19, 155, 42, 30, 77, 121, 98, 130, 9, 136, 142, 135, 2, 21, 107, 32, 157, 115, 162, 67]. Cette méthode est

largement utilisée car elle est simple à mettre en place mais elle ne permet pas d'établir une hiérarchie fiable entre les modèles, le résultat de l'analyse pouvant grandement dépendre des instances choisies ou de l'auteur lui-même.

Certains travaux proposent d'évaluer à quel point la carte met en valeur des indices visuels pertinents du point de vue de la tâche, i.e. l'objet à classer dans le cas de la classification d'image [127, 19, 155]. Une limite de cette méthode est qu'elle n'évalue pas seulement l'explication mais aussi le modèle. En effet, si un modèle utilise de mauvais indices visuels dans l'image, comme l'arrière-plan dans le cas de la détection d'objet, l'explication doit refléter ce comportement, i.e. mettre en valeur l'arrière-plan. Ainsi, une explication fidèle au comportement d'un mauvais modèle sera pénalisée par ce critère.

D'autres auteurs proposent des propriétés qu'il serait intéressant de respecter pour un modèle comme la cohérence et la fidélité locale [98], la sensibilité et la conservation [42] ou encore la sensibilité et l'invariance à l'implémentation [142]. Ces propriétés sont intéressantes parce qu'il est facile de vérifier si elles sont respectées ou non et parce qu'elles évaluent la fiabilité de l'explication et non pas la pertinence des indices utilisés par le modèle.

Dans la même optique, il a été proposé d'évaluer la fidélité des cartes de saillance à l'aide de métriques objectives dédiées que nous appelons ici *métriques de fiabilité* [2] [155, 19, 115, 77, 42]. Ces métriques proposent de perturber l'image traitée par le modèle afin de déterminer si les zones mises en valeur par l'explication contribuent effectivement de façon importante au score de la classe. Ces métriques évaluent la fidélité de l'explication par rapport au fonctionnement effectif du modèle. Elles n'ont donc pas vocation à remplacer une étude utilisateur, qui étudie la perception de l'explication par l'utilisateur, comme illustré en figure 1.17.

Dans la section suivante, nous introduisons toutes les métriques de fiabilité proposées jusqu'à maintenant dans la littérature à notre connaissance.

Deletion Area Under Curve [115] Pour évaluer la fiabilité d'une carte de saillance, Petsiuk et al. ont proposé une métrique appelée "aire sous la courbe en suppression" ("Deletion Area Under Curve", DAUC) [115]. Cette métrique évalue la fiabilité des cartes de saillance en masquant progressivement l'image en commençant par les zones les plus importantes selon la carte de saillance et en terminant par les moins importantes.

L'image d'entrée est un tenseur $I \in \mathbb{R}^{H \times W \times 3}$ et la carte de saillance est une matrice 2D $S \in \mathbb{R}^{H' \times W'}$ avec une résolution inférieure, $H' \leq H$ et $W' \leq W$. D'abord, S est parcouru dans l'ordre décroissant. À chaque élément $S_{i'j'}$, la zone correspondante de I est masquée en la

2. Nous utilisons aussi le terme de métriques de *fidélité* pour désigner ces métriques.

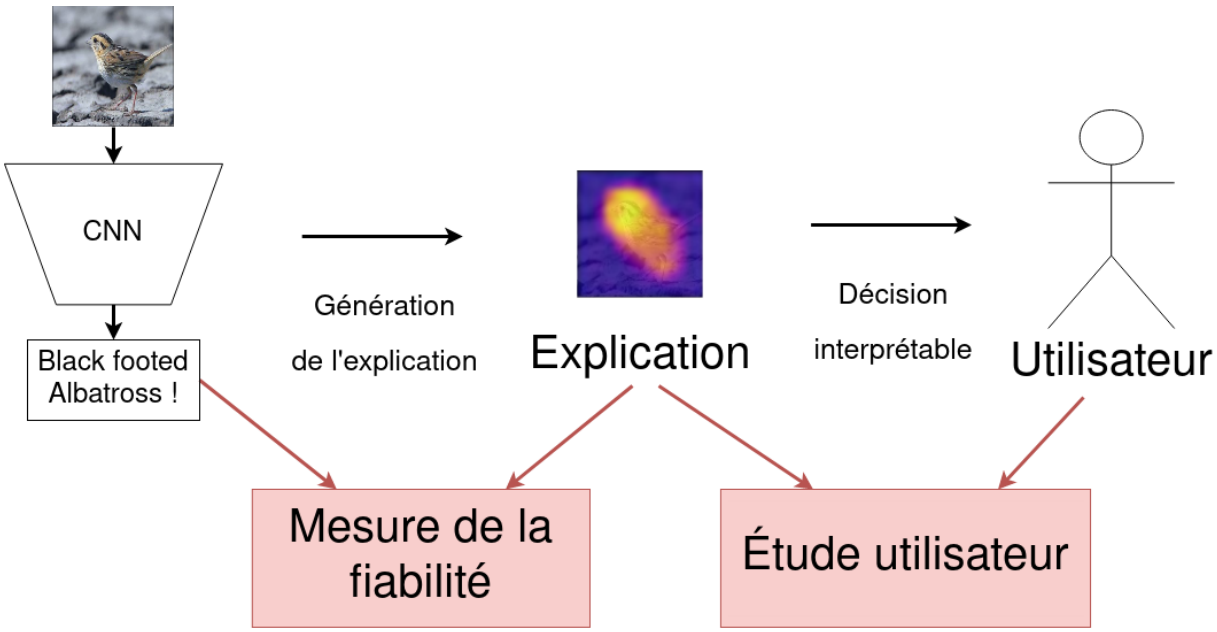


FIGURE 1.17 – Illustration du rôle des métriques de fiabilité et d'une étude utilisateur. Les métriques de fiabilité permettent d'évaluer la fiabilité d'une explication alors qu'une étude utilisateur étudie l'impact de l'explication sur la perception de l'utilisateur.

multipliant par un masque $M^{i',j'} \in \mathbb{R}^{H \times W}$, où

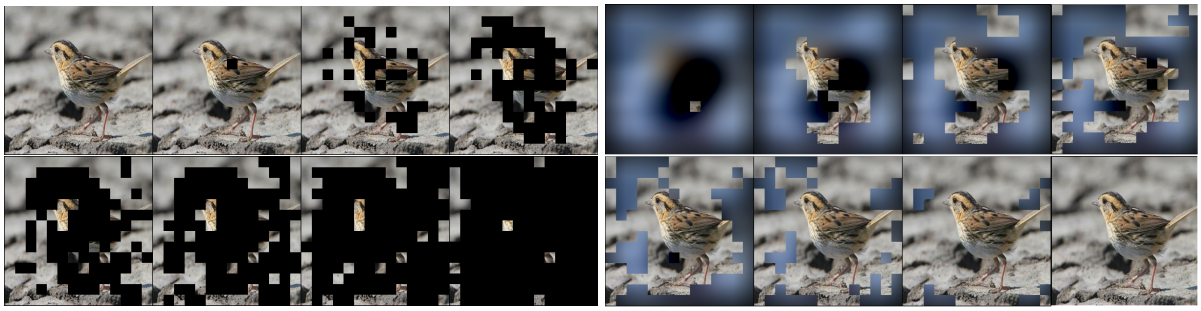
$$M_{ij}^{(i',j')} = \begin{cases} 0, & \text{si } i'r < i < i'(r+1) \text{ et } j'r < j < j'(r+1) \\ 1, & \text{sinon,} \end{cases} \tag{1.64}$$

où $r = H/H' = W/W'$. Des exemples d'images d'entrée obtenues lors de cette opération sont visibles dans la figure 1.18. Après la k-ième opération de masquage, le modèle m exécute une inférence avec la version actualisée de I, et le score de la classe initialement prédite est mis à jour, produisant un nouveau score c_k :

$$c_k = m(I \cdot \prod_{\tilde{k}=1}^{\tilde{k}=k} M^{(i_{\tilde{k}}, j_{\tilde{k}})}), \tag{1.65}$$

où $(i_{\tilde{k}}, j_{\tilde{k}})$ sont les indices de l'élément de de S masqué à l'étape $\tilde{k}$. Ensuite, une fois l'image entièrement masquée, les scores c_k sont normalisés en les divisant par le maximum $\underset{k}{max}\ c_k$. La métrique DAUC est finalement obtenue en calculant l'aire sous la courbe (AUC) des scores c_k normalisés en fonction de la proportion p_k de l'image qui est masquée. L'intuition est que si une carte de saillance met en évidence les zones pertinentes pour la décision, leur masquage

entraînera une diminution du score de classe initialement prédit, ce qui minimisera l'AUC. Par conséquent, la minimisation de cette métrique correspond à une amélioration.



(a) (b)

FIGURE 1.18 – Exemples d'images passées au modèle pendant le calcul de (a) DAUC et (b) IAUC.

Insertion Area Under Curve [115] Au lieu de masquer progressivement l'image, la métrique appelée "aire sous la courbe en ajout" ("Integration Area Under Curve",IAUC) part d'une image floutée, puis la rend progressivement nette en commençant par les zones les plus importantes selon la carte de saillance. De même, si les zones mises en valeur par la carte sont effectivement pertinentes pour la prédiction, le score de la classe initialement prédite (obtenu à partir de l'image non floutée) est censé augmenter rapidement. La maximisation de cette métrique correspond donc à une amélioration.

Increase In Confidence [19]. La métrique dite de l'"augmentation de la confiance" ("Increase in Confidence", IIC) mesure la fréquence à laquelle la confiance du modèle dans la classe prédite augmente lorsque l'on met en évidence les zones saillantes. Tout d'abord, l'image d'entrée est masquée avec la carte d'explication comme suit :

$$I_m = \text{norm}(\text{upsamp}(S)) \bullet I, \tag{1.66}$$

où $\text{norm}(S)$ est la fonction de normalisation min-max, définie comme $\text{norm}(S) = \frac{S - min(S)}{max(S) - min(S)}$, $\text{upsamp}(S)$ est une fonction qui augmente la résolution de S à la résolution de I, et $\bullet$ est le produit élément par élément. La métrique IIC est définie comme suit :

$$\text{IIC} = \mathbf{1}_{[c_I < c_{I_m}]}, \tag{1.67}$$

où c_I est le score de la classe prédite avec I en entrée et c_{I_m} est le score de la même classe avec I_m en entrée. L'intuition est qu'une carte de saillance S fiable met en évidence des zones telles que, lorsque les zones non saillantes sont supprimées, le score de la classe augmente. Par conséquent, la maximisation de cette métrique correspond à une amélioration. Notez que cette métrique est une valeur binaire et qu'elle n'est utile que pour calculer sa valeur moyenne sur un grand nombre d'images, comme ce sera fait dans le chapitre 2.

Average Drop [19]. Comme l'IIC, la métrique de "baisse moyenne" ("Average Drop", AD) mesure la baisse moyenne du score lors de la mise en évidence des zones saillantes. En utilisant le même masquage de l'image d'entrée, la métrique AD calcule la différence de score relative entre les deux images I et I_m :

$$AD = \frac{max(0, c_I - c_{I_m})}{c_I} \tag{1.68}$$

Étant donné qu'en mettant en évidence les zones saillantes, le score de classe n'est pas censé diminuer, la minimisation de cette métrique correspond à une amélioration. Comme le nom l'indique, cette métrique est obtenue en calculant une valeur moyenne sur plusieurs images, comme cela sera fait dans le chapitre 2. Cependant, nous n'avons pas inclus l'opérateur de calcul de la moyenne dans la définition de l'AD afin de conserver une notation simple et uniforme tout au long de cette section.

Average Drop in Deletion (ADD) [77]. Jung et al. ont proposé une variante de la métrique précédente qui consiste à masquer les zones saillantes au lieu des zones non saillantes. Pour ce faire, l'image est masquée avec l'inverse de la carte de saillance :

$$I_{1-m} = (1 - \text{norm}(\text{upsamp}(S))) \bullet I \tag{1.69}$$

La métrique ADD est alors définie comme suit :

$$ADD = \frac{max(0, c_I - c_{I_{1-m}})}{c_I} \tag{1.70}$$

Contrairement à AD, cette métrique élimine les zones saillantes et le score de classe devrait diminuer, ce qui signifie que la maximisation de cette métrique entraîne une amélioration. De même, comme pour AD, nous avons abandonné l'opérateur de calcul de la moyenne pour garder une notation simple.

À retenir

— Pour générer des cartes de saillances, il existe deux approches : les méthodes post-hoc et les modèles d'attention.
— Les méthodes post-hoc se divisent en trois catégories : les méthodes par perturbation, par pondération des cartes de caractéristiques et par rétropropagation vers l'image d'entrée.
— Les modèles d'attention se divisent en trois catégories : les attention convolutionelles, prototypiques et non paramétriques.
— La plupart des modèles d'attention utilisent des attentions prototypiques ou convolutionelles, i.e. paramétriques.
— Les métriques de fiabilité sont des métriques objectives qui évaluent la fiabilité des cartes de saillance en perturbant les zones de l'image mises en valeur par l'explication.

CHAPITRE 2

MÉTRIQUES POUR L'ÉVALUATION DES CARTES DE SAILLANCE

Dans ce chapitre nous étudions des métriques développées pour évaluer la fiabilité de cartes de saillance produites pour expliquer des modèles profonds de classification d'images. D'abord nous étudions des limites de deux métriques proposées précédemment dans la littérature, puis nous proposons deux nouvelles métriques pour pallier une de ces difficultés. Enfin, nous discutons des métriques étudiées dans ce chapitre et de leur rôle potentiel au sein d'une étude utilisateur.

2.1 Introduction

Ces dernières années ont vu un regain d'intérêt pour l'apprentissage automatique interprétable, car de nombreux modèles d'apprentissage de pointe sont actuellement des modèles profonds et souffrent de leur manque d'interprétabilité en raison de leur nature "boîte noire". En classification d'images, de nombreuses approches génériques ont été proposées pour expliquer la décision d'un modèle en générant des cartes de saillance qui mettent en évidence les zones importantes de l'image concernant la tâche à accomplir [127, 19, 115, 156, 2, 135, 77, 99]. En effet, la communauté de l'apprentissage profond interprétable n'a pas encore trouvé de consensus sur la manière d'évaluer les explications produites par le modèle, la principale difficulté résidant dans l'ambiguïté du concept d'interprétabilité. Selon le contexte applicatif, les exigences des utilisateurs en termes d'interprétabilité peuvent varier fortement, rendant difficile la recherche d'un protocole d'évaluation universel. C'est ainsi qu'est née une tendance dans la littérature où les auteurs confrontent les utilisateurs aux décisions des modèles accompagnées d'explications afin de déterminer la préférence des utilisateurs sur une application particulière [6, 144, 147]. Les principaux problèmes de cette approche sont son coût financier et la difficulté d'établir un protocole correct. En effet, il faut parvenir concevoir une expérience dont les résultats aideront à comprendre les besoins des utilisateurs et ce malgré que la plupart des chercheurs en apprentissage automatique ne sont pas formés à mener des expériences impliquant des humains.

En raison de ces problèmes, une autre tendance propose de concevoir des métriques objectives pour évaluer la fiabilité des méthodes d'explication génériques [115, 77, 19]. Ces métriques consistent à évaluer la fiabilité d'une méthode en vérifiant si la saillance des zones de l'image correspond bien à l'impact réel qu'elles ont sur le score de classification. Dans cet article, nous avons choisi de suivre cette tendance, en étudiant les métriques DAUC et IAUC proposées par [115] et en proposant trois nouvelles métriques. Tout d'abord, nous discutons de deux limites de ces métriques. Premièrement, pendant le calcul de DAUC et IAUC, le modèle traite des images qui sont hors de la distribution d'entraînement, ce qui pourrait conduire à un comportement inattendu du modèle et donc de la méthode utilisée pour générer les cartes de saillance. Deuxièmement, les valeurs des scores de saillance données par la carte de saillance sont ignorées par ces métriques car elles ne prennent en compte que le classement des scores. Cela montre que ces métriques sont insuffisantes en elles-mêmes, car l'apparence visuelle d'une carte de saillance peut changer de manière significative sans que le classement des scores ne soit modifié. Pour pallier ce second problème, nous proposons trois nouvelles métriques qui prennent en compte les scores. La première métrique que nous appelons *parcimonie* est conçue pour quantifier à quel point une carte de saillance est concentrée sur un ou des points spécifiques de l'image, une propriété ignorée par les travaux précédents. Les deux autres sont appelées "corrélation de suppression" ("Deletion Correlation", DC) et "corrélation d'insertion" ("Insertion Correlation", IC) et mesurent la calibration des cartes de saillance, une quantité également ignorée jusqu'à présent. Enfin, nous donnons des remarques générales sur les métriques étudiées dans ce chapitre et discutons de leur rôle potentiel au sein d'une étude utilisateur.

2.2 Limites des métriques DAUC et IAUC

DAUC et IAUC génèrent des images hors de la distribution d'entraînement. Lors du masquage/débrouillage progressif de l'image d'entrée, le modèle reçoit des échantillons qui peuvent être considérés comme "hors de la distribution" ("Out Of Distribution", OOD). En effet, les distorsions produites par les opérations de masquage et de flou n'existent pas dans les images vues par le modèle pendant l'entraînement, où on utilise des augmentations standards comme le recadrage aléatoire, le retournement horizontal et la modification des couleurs, comme illustré en figure 2.1. Cela signifie que le modèle n'a pas appris à traiter des images présentant de telles distorsions.

Par conséquent, la distribution des images présentées au modèle lors du calcul de DAUC/IAUC est probablement différente de celle rencontrée lors de l'entraînement. De plus, il a été documenté

(a) Entraînement (b) DAUC/IAUC

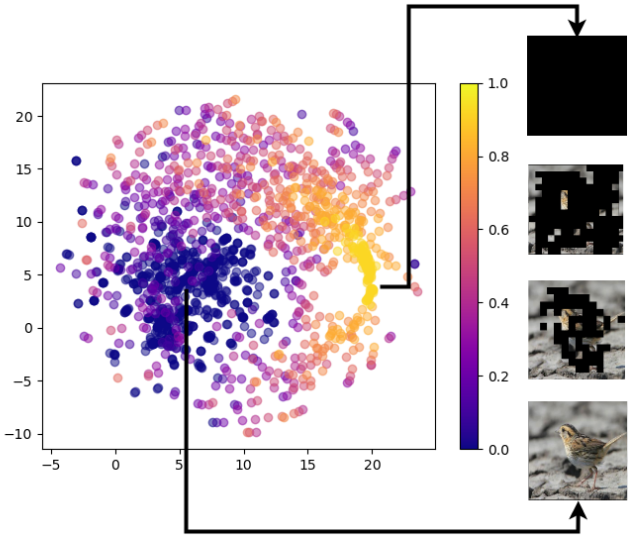
FIGURE 2.1 – Les transformations vues par le modèle pendant le calcul de DAUC/IAUC sont différentes des transformations auxquelles il a été confronté pendant l'entraînement.

que les CNNs et plus généralement les modèles d'apprentissage profond généralise mal en dehors de la distribution d'entraînement [46]. Ceci montre que DAUC et IAUC peuvent ne pas refléter la fidélité des méthodes d'explication car elles sont basées sur un comportement du modèle différent de celui rencontré face à la distribution d'entraînement (par exemple pendant la phase de test).

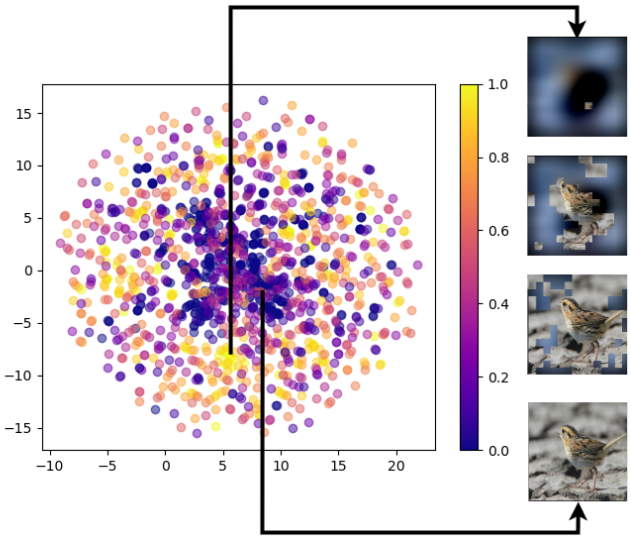
Pour vérifier cette hypothèse, nous visualisons les projections UMAP [103] des représentations de 100 échantillons masqués/floutés obtenues lors du calcul de DAUC et IAUC sur le jeu de données CUB-200-2011 [152]. Nous avons également ajouté la représentation de 500 images de test non modifiées (en bleu) pour visualiser la distribution de l'apprentissage. Le modèle entraîné est un ResNet50 [58] sur lequel nous avons appliqué Grad-CAM++ [19]. La figure 2.2 montre que, lors du calcul du DAUC, les représentations convergent progressivement vers un point unique, ce qui n'est pas surprenant puisqu'à la fin du calcul, toutes les images sont entièrement masquées, c'est-à-dire complètement noires. Cependant, même lorsque seulement 40% de l'image est masquée, la représentation correspondante est éloignée de la distribution d'entraînement (le nuage de points bleus). Un phénomène similaire se produit avec l'IAUC, où le fait de brouiller l'image fait que la représentation s'éloigne de la distribution d'entraînement.

Afin d'apporter une analyse quantitative, nous proposons d'évaluer la séparabilité des représentations des images produites par les métriques des représentations des images non modifiées du jeu de test. Pour cela, plusieurs modèles sont entraînés à distinguer les représentations issues d'images masquées/floutées des représentations issues d'images non modifiées. Si au moins un de ces modèles parvient à une performance significative, cela signifie que les deux groupes d'images sont séparables. Cette séparabilité confirmerait que les images avec des patchs flous ou noirs ont des représentations distinctes des images de la distribution d'entraînement, ce qui montrerait qu'elles sont bien OOD.

Nous entraînons une machines à vecteurs de support ("Support Vector Machine", SVM), un



(a)



(b)

FIGURE 2.2 – Projection UMAP des représentations obtenues en calculant (a) DAUC et (b) IAUC sur 100 images. La couleur indique la proportion de l'image qui est masquée/non floue, i.e. le nombre de patchs noirs/flous appliqué sur l'image originale. Le modèle entraîné est un ResNet50 sur lequel nous avons appliqué Grad-CAM++ sur le jeu de données CUB-200-2011. Nous avons également affiché les représentations de 500 points de l'ensemble de test pour visualiser la distribution de l'entraînement (en bleu). En masquant progressivement l'image, les représentations convergent vers un point (en jaune) qui est éloigné des points correspondant aux images sans patchs noirs (en bleu). De même, en brouillant l'image, la représentation s'éloigne de la distribution d'entraînement. L'opération de masquage/floutage crée donc effectivement des échantillons OOD.

Modèle	Détection de patchs noirs	Détection de patchs flous
SVM	0.933	0.858
KNN	0.804	0.604
DT	0.814	0.711
NN	0.984	0.156

TABLE 2.1 – Aire sous la courbe ROC de modèles entraînés à distinguer les représentations d'images contenant des patchs noirs/flous de représentations d'images non modifiées.

classifieur à plus proche voisin ("K-Nearest Neighbors", KNN), un arbre de décision ("Decision Tree", DT) et un perceptron multi-couches ("Multi-Layer Perceptron", MLP) à attribuer le label 1 aux représentations issues d'images avec des patchs noirs/flous et le label 0 aux représentations des images non modifiées. Les modèles sont entraînés et évalués à partir des représentations du même ResNet-50 expliqué par Grad-CAM++ produisant des cartes de saillance en résolution 14×14 que pour la figure 2.2. On utilise également les mêmes 100 images pour constituer le jeu de données. Pour chacune de ces images, il y a donc une représentation d'une image non modifiée et $14 \times 14 = 196$ représentations d'images modifiées sur lesquelles se trouvent entre 1 et 196 patchs noirs/flous. Au total, les classes 0 et 1 sont respectivement constituée de 1×100 et $196 \times 100 = 19600$ images. Le jeu de données est donc hautement déséquilibré et nous mesurons donc la performance par l'aire sous la courbe ROC des modèles évalués sur le jeu de test. Nous utilisons respectivement 40%, 10% et 50% du jeu de données pour l'entraînement, la validation et le test. L'entraînement et l'évaluation est réalisée avec la librairie `scikit-learn` (version 1.0.2) en utilisant les valeurs par défaut des paramètres. Les résultats sont visibles en tableau 2.1. On constate que les modèles atteignent des performances non négligeables, voir élevées pour le modèle SVM, ce qui montre que les images modifiées sont bien OOD. Notez que les performances sont plus faibles avec les patchs flous (colonne IAUC) qu'avec les patchs noirs (colonne DAUC) ce qui indique que les échantillons produits par IAUC sortent moins de la distribution que ceux produits par DAUC, ce qui est cohérent avec l'observation en figure 2.2.

Ces deux expériences démontrent que les métriques DAUC et IAUC présentent effectivement des échantillons OOD, ce qui pourrait conduire à un comportement inattendu du modèle et de la méthode utilisée pour générer les cartes d'explication. Cependant, comme le soupçonne Petsiuk et al. [115], l'opération de flou semble créer des échantillons moins éloignés de la distribution d'entraînement par rapport à l'opération de masquage, probablement parce qu'une image floue contient toujours les composantes en basse fréquence de l'image originale. Il faut aussi noter que

la plupart des modèles de classification actuels sont conçus en supposant qu'une image d'entrée contient un objet à reconnaître, ce qui est en contradiction avec DAUC et IAUC puisqu'ils consistent à supprimer l'objet à reconnaître de l'image. Cela suggère que la modification de ces métriques de manière à toujours laisser un objet à reconnaître dans l'image d'entrée pourrait résoudre ce problème.

DAUC et IAUC ne prennent en compte que le rang du score du pixel. Lors du calcul de DAUC et IAUC, la carte de saillance est utilisée uniquement pour déterminer dans quel ordre masquer/révéler l'image d'entrée. Par conséquent, seul le classement des scores de saillance S_{ij} est utilisé pour déterminer dans quel ordre masquer l'image, les valeurs des scores étant ignorées.

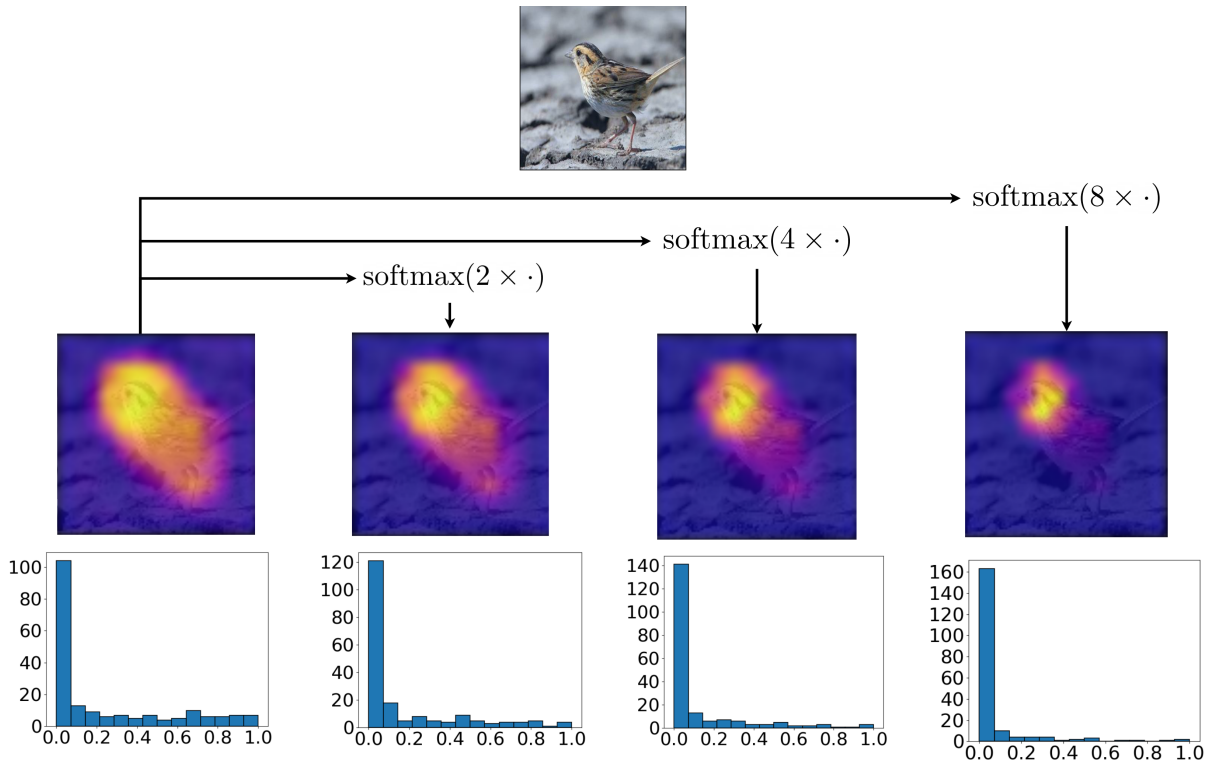
Toutefois, le classement des pixels n'est pas la seule caractéristique à prendre en compte, car l'aspect visuel peut varier considérablement entre deux cartes de saillance sans que le classement ne soit modifié. La figure 2.3a montre des exemples de carte de saillance dont la distribution des scores a été modifiée artificiellement. Nous avons utilisé une carte de saillance produite par la méthode d'explication Score-CAM [156] et modifié la distribution de ses scores en multipliant toutes les valeurs par un coefficient suivi de l'application d'une fonction softmax. En augmentant le coefficient, nous modifions l'aspect visuel des cartes, sans changer l'ordre des pixels, qui conservent donc les mêmes scores DAUC et IAUC. Cela illustre le fait que DAUC et IAUC ne tiennent pas compte de la dynamique des scores de la carte de saillance, qui peut affecter considérablement l'aspect visuel. Pour compléter DAUC et IAUC, nous proposons dans la section suivante de nouvelles métriques qui prennent en compte les valeurs de score.

2.3 Métriques prenant en compte le score

Comme mentionné dans le paragraphe précédent, DAUC et IAUC ignorent les valeurs réelles des scores et ne prennent en compte que le classement des scores de saillance S_{ij}. Pour compléter ces métriques, nous proposons trois nouvelles métriques, appelées "Parcimonie", "corrélation de suppression" ("Deletion Correlation", DC) et "corrélation d'insertion" ("Insertion Correlation", IC).

2.3.1 La métrique de parcimonie

Un aspect visuel important des cartes de saillance qui n'a pas été étudié jusqu'à présent par la communauté est une propriété que nous appelons la parcimonie. Comme le montre la



(a)

Modèle	Visualisation	Transformation	DAUC	IAUC	Parcimonie	DC	IC
ResNet50	Score-CAM	None	0.012	0.52	3.98	0.187	0.22
		softmax$(2 \times \cdot)$			5.96	0.24	0.19
		softmax$(4 \times \cdot)$			8.96	0.29	0.14
		softmax$(8 \times \cdot)$			16.52	0.38	0.04

(b)

FIGURE 2.3 – (a) Exemples de cartes de saillance obtenues en modifiant les scores d'une carte de saillance pour changer son apparence visuelle. La carte de saillance originale est générée à l'aide de Score-CAM appliqué à un modèle ResNet50 testé sur le jeu de données CUB-200-2011. (b) Malgré des apparences visuelles différentes, les valeurs des métriques DAUC et IAUC sur ces quatre cartes sont identiques, car ces métriques ignorent les valeurs des scores et ne prennent en compte que le classement des scores. En revanche, la métrique de parcimonie dépend de la distribution des scores et reflète le degré de focalisation de la carte. Dans cette figure, seule la carte de saillance est modifiée, le processus de décision reste inchangé.

figure 2.3, les cartes de saillance peuvent être plus ou moins focalisées sur un point spécifique en fonction de la distribution des scores, sans modifier le classement des scores. Cet aspect pourrait avoir une incidence sur l'interprétabilité de la méthode, car il modifie considérablement l'aspect visuel de la carte et pourrait donc également affecter la perception de l'utilisateur. Par exemple, il pourrait être argumenté qu'une carte de saillance avec une valeur de parcimonie élevée implique une carte dont l'activation se trouve sur un endroit précis et ne met en évidence que quelques éléments de l'image d'entrée, ce qui la rend plus facile à comprendre pour les humains. Cette métrique est définie comme suit :

$$\text{Parcimonie} = \frac{S_{max}}{S_{mean}}, \tag{2.1}$$

où S_{max} et S_{mean} sont respectivement le score maximum et le score moyen de la carte de saillance S. Notez que les méthodes de saillance disponibles dans la littérature génèrent des cartes de saillance dont les scores peuvent être compris dans des intervalles variés. Par conséquent, la carte doit d'abord être normalisée comme suit :

$$S' = \frac{S - S_{min}}{S_{max} - S_{min}}. \tag{2.2}$$

Cela signifie que, après normalisation, $S'_{max} = 1$ et que l'équation (2.1) peut être simplifiée à

$$\text{Parcimonie} = \frac{1}{S'_{mean}}. \tag{2.3}$$

Une valeur de parcimonie élevée signifie un rapport S_{max}/S_{moyen} élevé, c'est-à-dire un score moyen faible S_{moyen} qui indique que les zones activées de la carte sont petites et focalisées. Comme le montre le figure 2.3b, cette métrique est sensible aux valeurs des scores de saillance et reflète les différents degrés de concentration observés dans les cartes de saillance.

2.3.2 Les métriques DC et IC

Comme mentionné précédemment, les métriques DAUC et IAUC ignorent les valeurs de score des cartes de saillance et ne prennent en compte que le classement des scores. Cela signifie que ces mesures ignorent la parcimonie de la carte, c'est pourquoi nous avons proposé de quantifier cet aspect. Une autre propriété potentiellement intéressante des cartes de saillance est leur calibration. Le concept de calibration a connu un récent regain d'intérêt dans la communauté de l'apprentissage profond [53, 169, 110], mais les travaux précédents se sont concentrés exclusivement sur la calibration des scores de prédiction. Un pixel S_{ij} d'une carte de saillance

S bien calibrée refléterait par sa luminosité l'importance qu'il a sur le score de classe. Plus précisément, on dit qu'une carte explicative S est parfaitement calibrée si pour deux éléments quelconques S_{ij} et $S_{i'j'}$, on a $S_{ij}/S_{i'j'} = v/v'$, où v et v' sont respectivement l'impact de S_{ij} et $S_{i'j'}$ sur le score de classe. Pour évaluer cela, nous proposons de quantifier la corrélation entre les scores de saillance et leur impact correspondant sur le score de classe. À notre connaissance, c'est la première fois qu'une métrique objective est proposée pour mesurer la calibration des méthodes d'explication. Contrairement aux travaux qui étudient les prédictions du modèle pour voir si les scores de classe reflètent la probabilité que le modèle commettent une erreur, [53, 169, 110], cette métrique étudie le lien entre les scores de saillance des zones de l'image et l'impact effectif de ces zones sur le score de classe.

Nous nous inspirons des métriques DAUC et IAUC et proposons de masquer/révéler progressivement l'image d'entrée en suivant l'ordre suggéré par la carte de saillance, mais au lieu de calculer l'aire sous la courbe du score de classe par rapport au rang du pixel, nous calculons la corrélation linéaire des variations du score de classe et des scores de saillance. La corrélation mesurée lors du masquage de l'image est appelée "corrélation de suppression" ("Deletion Correlation", DC) et celle mesurée lors de la révélation de l'image est appelée "corrélation d'insertion" ("Insertion Correlation", IC).

La métrique DC est calculée en utilisant la même méthode de masquage et inférence successifs que DAUC. Une fois les scores c_k calculés, nous calculons la variation des scores $v_k = c_k - c_{k+1}$. Enfin, nous calculons la corrélation linéaire entre les v_k et les s_k où s_k est le score de saillance de la zone masquée à l'étape k. Pour la métrique IC, nous nous inspirons de IAUC, et au lieu de masquer l'image, nous partons d'une image floue, et révélons progressivement l'image en fonction de la carte de saillance. Une fois l'image totalement révélée, les variations de score sont calculées $v_k = c_{k+1} - c_k$ et nous calculons la corrélation linéaire des v_k avec les s_k. Notez que l'ordre de la soustraction est inversé par rapport au DC, car lors de la révélation de l'image, le score de la classe est censé augmenter.

L'intuition est que lorsque l'on calcule DC/IC sur une méthode de saillance bien calibrée, lorsque la variation du score de classe est élevée, le score de saillance est également élevé, et inversement, lorsque la variation du score de classe est faible, le score de saillance devrait également être faible. Les métriques DC et IC mesurent la calibration, un aspect qui est ignoré par les métriques DAUC et IAUC mais aussi par la métrique de parcimonie. Les métriques DC et IC sont calculées sur des exemples en figure 2.3.

2.3.3 Limites

La métrique de parcimonie ne prend pas en compte les scores de prédiction. En effet, cette métrique ne considère que la dynamique du score de saillance et ignore le score de classe produit par le modèle. Cependant, cette métrique est conçue pour être utilisée en complément d'autres métriques comme DAUC, IAUC, DC ou IC, qui prennent en compte le score de classe.

Les métriques DC et IC génèrent également des images OOD. En effet, DC et IC utilisent le même procédé de masquage/flou que DAUC et IAUC, on peut donc avancer la fiabilité de DC et IC pourrait être améliorée en n'utilisant pas d'images OOD.

2.4 Base de référence

Nous calculons les cinq métriques étudiées dans ce chapitre (DAUC, IAUC, DC, IC et Parcimonie) sur des méthodes d'explication génériques post-hoc et des architectures d'attention qui intègrent le calcul de la carte de saillance dans leur inférence. Les méthodes post-hoc sont Grad-CAM [127], Grad-CAM++ [19], RISE [115], Score-CAM [156], Ablation CAM [31] et la méthode de base de la carte d'activation (AM). Cette dernière méthode consiste à simplement visualiser la norme euclidienne du vecteur de caractéristiques de la dernière couche. Les architectures avec attention sont B-CNN [64], le modèle de [67] que nous appelons IBP (abréviation de Interpretability By Parts), ProtoPNet [21], et ProtoTree [107]. Nous utilisons également BR-NPA, une architecture d'attention spatiale non paramétrique que nous proposons et qui sera introduite dans le chapitre suivant. Ces modèles sont dotés de plusieurs têtes d'attention et peuvent donc se concentrer sur plusieurs endroits différents de l'image. Chaque tête génère une carte de saillance (ou carte d'*attention*) et le modèle produit donc plusieurs cartes pour une image d'entrée. Cependant, les métriques sont conçues pour évaluer des approches avec une seule carte de saillance par image. Étant donné que dans ces architectures, la première carte d'attention est la plus importante pour la décision, nous proposons donc de n'utiliser que celle-ci pour calculer les métriques.

Détails d'implémentation. La colonne vertébrale [1] utilisée pour tous les réseaux est ResNet-50 [58]. Les images sont augmentées pendant l'entraînement en utilisant un recadrage aléatoire de 448×448 et un retournement horizontal aléatoire. Pendant le test, nous extrayons un recadrage

1. Traduction de l'anglais "backbone".

Modèle	Visualisation	Précision	DAUC↓	IAUC↑	DC↑	IC↑	Parcimonie↑
ResNet50	Ablation CAM	84.2	0.0215	0.26	0.36	−0.04	8.54
	Grad-CAM		0.0286	0.16	0.35	−0.12	5.28
	Grad-CAM++		0.0161	0.21	0.35	−0.07	6.73
	RISE		0.0279	0.18	**0.57**	−0.11	6.63
	Score-CAM		0.0207	0.27	0.32	−0.05	5.96
	AM		0.0362	0.22	0.31	−0.09	4.04
B-CNN	-	84.8	0.0208	0.3	0.27	−0.02	12.74
BR-NPA		**85.5**	**0.0155**	**0.49**	0.41	−0.02	**16.02**
IBP		81.9	0.0811	0.48	0.23	−0.04	6.56
ProtoPNet		84.8	0.2964	0.37	0.1	−0.06	2.18
ProtoTree		82.1	0.2122	0.43	0.17	**0.04**	13.75

TABLE 2.2 – Évaluation de l'interprétabilité sur le jeu de données CUB-200-2011.

central de taille 448×448. Nous utilisons 10% des images de l'ensemble d'apprentissage pour la validation. Les modèles sont entraînés en utilisant l'entropie croisée pendant 10 époques. Le meilleur modèle sur l'ensemble de validation est restauré pour la phase de test. Les hyperparamètres suivants ont été recherchés sur l'ensemble de validation en utilisant la bibliothèque python Optuna [5] avec l'échantillonneur par défaut (un algorithme de Parzen Estimator structuré en arbre) : le taux d'apprentissage, le *momentum*, l'optimiseur, la taille du lot, le *dropout* sur la couche de classification et le *weight decay*. Nous utilisons Pytorch 1.10.2 [26] et deux GPU P100. En suivant le travail original de Petsiuk et al. [115], nous avons échantillonné 4000 masques à une résolution de 7×7 pour la méthode post-hoc RISE. Il faut noter que la manière dont l'image est floutée n'est pas précisée par Petsiuk et al. lors de l'introduction de la métrique IAUC [115]. Nous proposons donc d'appliquer un noyau de convolution de taille 121×121 dont les valeurs sont toute égales à $1/(121 \times 121)$ en complétant l'image originale avec des valeurs nulles pour que l'image floutée résultante soit de même taille.

Le tableau 2.2 montre les performances obtenues. Il faut noter les faibles valeurs de corrélation obtenues, en particulier pour IC, où la plupart des valeurs sont très proches de 0, ce qui signifie que les scores de saillance ne reflètent pas l'impact sur le score de la classe. Cela souligne le fait que les modèles d'attention et les méthodes d'explication ne sont actuellement pas conçus pour cet objectif, bien que cela puisse être une propriété intéressante.

2.5 Discussion

Une limite commune à toutes les métriques évoquées ici est qu'elles sont conçues pour des méthodes et des architectures produisant une seule carte de saillance par image, ce qui rend leur utilisation pour des architectures générant plusieurs cartes d'attention comme B-CNN, BR-NPA, IBP, ProtoPNet et ProtoTree peu adaptée. Dans la tableau 2.2, nous avons choisi de ne sélectionner que la carte d'attention la plus importante, mais les autres devraient également être prises en compte pour refléter pleinement le comportement du modèle. Nous aurions également pu calculer la moyenne des cartes d'attention à partir de toutes celles produites par le modèle mais cela ne serait pas non plus fidèle au modèle. En effet, cela reviendrait à considérer que toutes les cartes d'attention ont le même poids dans la décision, ce qui n'est pas vrai, puisque la première carte d'attention a plus d'importance dans la décision que la deuxième, qui est plus importante que la troisième, etc. Une possibilité serait d'estimer le poids de chaque carte et de calculer une moyenne pondérée, mais il reste le problème du calcul du poids, pour lequel il faudrait sans doute trouver une méthode empirique en raison de la variété des architectures.

Notez que les faibles valeurs de DC et IC dans la tableau 2.2 n'impliquent pas que les modèles et méthodes fournissent des performances insatisfaisantes, mais montrent simplement que la propriété de calibration n'a pas été étudiée jusqu'à présent. Enfin, soulignons que contrairement à DAUC, IAUC, DC et IC, la parcimonie n'évalue pas la fiabilité des cartes de saillance. En quantifiant un aspect visuel des cartes, l'utilité de cette métrique se situe dans l'étude du lien entre l'explication et la manière dont elle est perçue par l'utilisateur (cf. figure 1.17 du chapitre 1). En effet, on peut supposer qu'un utilisateur perçoive différemment une carte en fonction de son niveau de parcimonie.

2.6 Conclusion

Dans ce chapitre, nous avons d'abord étudié deux limites des métriques DAUC et IAUC. Nous avons montré qu'elles peuvent générer des échantillons OOD, ce qui peut avoir un impact négatif sur leur fiabilité. Nous montrons également qu'elles ne prennent en compte que le classement des scores de saillance et que l'apparence visuelle d'une carte de saillance peut changer de manière significative sans que les métriques DAUC et IAUC soient affectées. Ensuite, nous proposons de quantifier deux aspects qui n'ont pas été étudiés jusqu'à présent sur les cartes de saillance, la parcimonie et la calibration (les métriques DC et IC). Ces travaux ont fait l'objet d'une publication dans une conférence internationale avec comité de lecture [48].

À retenir

— Les métriques DAUC et IAUC mesurent la fiabilité d'une carte de saillance en masquant ou en révélant progressivement l'image et en mesurant l'évolution du score de la classe.
— Ces métriques ignorent les scores de saillance et ne tiennent compte que du tri des pixels de la carte de saillance.
— Nous introduisons une métrique pour mesurer la parcimonie, ainsi que les métriques DC et IC pour mesurer la calibration de la carte de saillance, des propriétés ignorées jusqu'à présent.

CHAPITRE 3

ATTENTION NON PARAMÉTRIQUE BILINÉAIRE REPRÉSENTATIVE

Dans ce chapitre nous décrivons un nouveau modèle que nous proposons, appelé "Attention non paramétrique bilinéaire représentative" ("Bilinear Representative Non-Parametric Attention", BR-NPA). Ce modèle est basé sur le modèle B-CNN introduit précédemment. D'abord sont décrites les modifications faites à B-CNN pour obtenir BR-NPA : l'augmentation de la résolution des cartes d'attention et le remplacement du module d'attention par un module non paramétrique. Ensuite, BR-NPA est évalué sur plusieurs tâches de classification telles que la classification à grain fin, la réidentification de personnes et la classification avec peu d'exemple. Enfin, l'interprétabilité de BR-NPA est évaluée en utilisant les métriques introduites dans le chapitre précédent.

3.1 Le modèle BR-NPA

Le modèle d'attention proposé se compose (1) de l'extraction de cartes de caractéristiques à haute résolution, (2) de la génération de vecteurs de caractéristiques représentatifs obtenus par regroupement de vecteurs similaires, et (3) de la concaténation des vecteurs de caractéristiques représentatifs et de leur passage à la couche de classification. La sortie d'un réseau convolutif, un tenseur de taille $C \times H \times W$, est appelé *volume de caractéristiques* ou *cartes de caractéristiques*. Une fois qu'une position spatiale (h, w) est choisie, où $h \in [1, H]$ et $w \in [1, W]$, un vecteur de dimension C, le *vecteur de caractéristiques*, peut être extrait.

3.1.1 Des cartes de caractéristiques haute-résolution

Avec un réseau CNN standard, la résolution de la dernière carte de caractéristiques est généralement beaucoup plus faible que celle de l'entrée. Par exemple, après avoir passé une image de taille 448×448 à un réseau ResNet-50 [58], la taille de la carte de caractéristiques

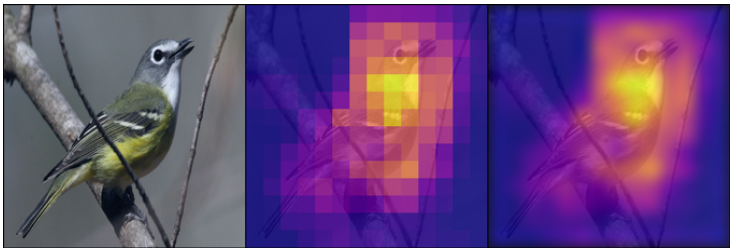


FIGURE 3.1 – L'effet de l'interpolation sur la visualisation d'une carte d'attention. De gauche à droite : une image du jeu de données CUB-200-2011, une carte de saillance produite par Grad-CAM++ pour expliquer un modèle ResNet-50 sans interpolation et la même carte de saillance avec interpolation bi-cubique. L'interpolation permet de donner une apparence plus douce mais n'augmente qu'artificiellement la résolution et ne permet donc pas d'avoir des cartes plus détaillées.

de sortie est réduite à 14×14. Il est donc difficile d'interpréter la manière dont le réseau neuronal a été activé par rapport à l'image, car chaque vecteur de caractéristiques couvre une zone importante et potentiellement hétérogène de l'image, aussi bien l'objet que l'arrière-plan. Lorsque les cartes de caractéristiques sont en haute résolution, elles ont un champ réceptif plus petit, et chaque vecteur de caractéristiques correspond à une zone réduite, plus susceptible d'être homogène d'un point de vue sémantique. Pour augmenter la résolution de la carte d'attention, des interpolations bi-linéaires ou bi-cubiques sont couramment utilisées pour la visualisation [127, 114, 34, 140]. Néanmoins, ces opérations conduisent généralement à des cartes d'attention floues et sans détails comme illustré en figure 3.1.

Une autre alternative consiste à augmenter la taille des images d'entrée, comme le fait [45]. Cependant, cette méthode augmente inévitablement le temps de calcul de l'inférence. Au lieu de cela, nous proposons de modifier l'architecture du CNN utilisé pour augmenter la résolution spatiale des cartes de caractéristiques. Par exemple, pour la classification fine, les pas des couches de sous-échantillonnage des étages 3 et 4 de ResNet-50 [58] sont réduits de 2 à 1. En procédant ainsi, la résolution des cartes de caractéristiques est passée de 14×14 à 56×56 (cf. section 3.2.1). Cela limite l'augmentation du coût de calcul puisque seules les cartes de caractéristiques des dernières couches deviennent plus grandes.

Dans toutes nos expériences, les modèles sont préentrainés (par exemple sur ImageNet [122] pour la classification fine) et cette modification implique que les noyaux des dernières couches seront appliqués à une échelle pour laquelle ils n'ont pas été préentrainés.

En effet, réduire les pas de sous-échantillonnage va confronter les noyaux des couches 3 et 4 à des cartes de traits avec une résolution plus importante et à un plus grand niveau de détail que celui auquel ils ont été entraînés.

Par conséquent, au cours de l'apprentissage, le modèle doit apprendre des caractéristiques adaptées à la tâche mais aussi à la nouvelle échelle, ce qui entraîne un temps d'apprentissage plus long et réduit l'efficacité de la procédure d'apprentissage par transfert. Pour pallier ce problème, la méthode de distillation proposée dans [60] est employée. Plus précisément, un réseau élève en haute résolution (56×56) est formé pour imiter un réseau professeur avec la résolution par défaut (14×14). Lors de l'entraînement du réseau élève, la fonction de coût d'entropie croisée habituelle a été utilisée conjointement avec la divergence KL entre les prédictions du réseau de l'élève et du réseau professeur :

$$L = \frac{1}{N} \sum_{i} \alpha \mathrm{CE}(\tilde{y}_s, y) + (1 - \alpha)\mathrm{KL}(\tilde{y}_t || \tilde{y}_s), \tag{3.1}$$

où $\tilde{y}_t$, $\tilde{y}_s$ sont respectivement la sortie du réseau professeur et élève, y est la vérité terrain et α est un paramètre qui équilibre le terme d'entropie croisé et le terme de divergence KL. Il faut noter que nous avons également utilisé l'architecture HR-Net [141] qui fournit des cartes de caractéristiques de 56×56, mais avons obtenu des performances faibles, probablement en raison du nombre réduit de cartes de caractéristiques haute résolution fournies par HR-Net (16 à 64 selon le nombre de couches, alors que ResNet [58] fournit 512 à 2048 cartes de caractéristiques).

3.1.2 Une couche d'attention non paramétrique

Dans la plupart des cas, l'objet d'intérêt est composé de plusieurs parties, où chaque partie joue potentiellement un rôle différent dans la classification. Dans [92], les caractéristiques correspondant à chaque partie d'objet ont été agrégées séparément pour construire une matrice de caractéristiques, c'est-à-dire une liste de vecteurs de caractéristiques. La séparation des parties locales de l'objet est réalisée via des couches d'attention composées de multiples blocs convolutifs [64], qui sont paramétriques.

Cependant, il est possible d'obtenir de bonnes performances sur les mêmes tâches de classification avec un CNN sans couche d'attention. Par exemple avec un ResNet-50 on obtient 84.2% de précision sur CUB-200-2011 (cf. tableau 3.6) ce qui est similaire aux performances des modèles d'attention qui sont comprises entre 81 et 85% de précision (cf. tableau 3.1). Cela signifie que les couches de convolutions d'un CNN sans attention suffisent au moins en partie pour trouver les parties importantes et les séparer les unes des autres. Il est donc intuitivement intéressant d'explorer les similarités entre les vecteurs de caractéristiques pour identifier des parties de l'objet et de visualiser les similarités entre les vecteurs comme une "attention".

Au lieu d'incorporer le module neuronal d'attention habituel, nous proposons de former

séquentiellement une liste de vecteurs de caractéristiques raffinés en regroupant les vecteurs de caractéristiques en fonction de leur *activation* et de leur similarité les uns avec les autres. Dans cette étude, le niveau d'*activation* a d'un vecteur de caractéristiques f est quantifié par sa norme euclidienne, c'est-à-dire $||f||^2$. Le vecteur caractéristique f, dont la norme est l'une des plus élevées dans le volume caractéristique correspondant, est considéré comme *actif*. En outre, un vecteur de caractéristiques est considéré comme singulier si les autres caractéristiques sélectionnées ne lui sont pas similaires.

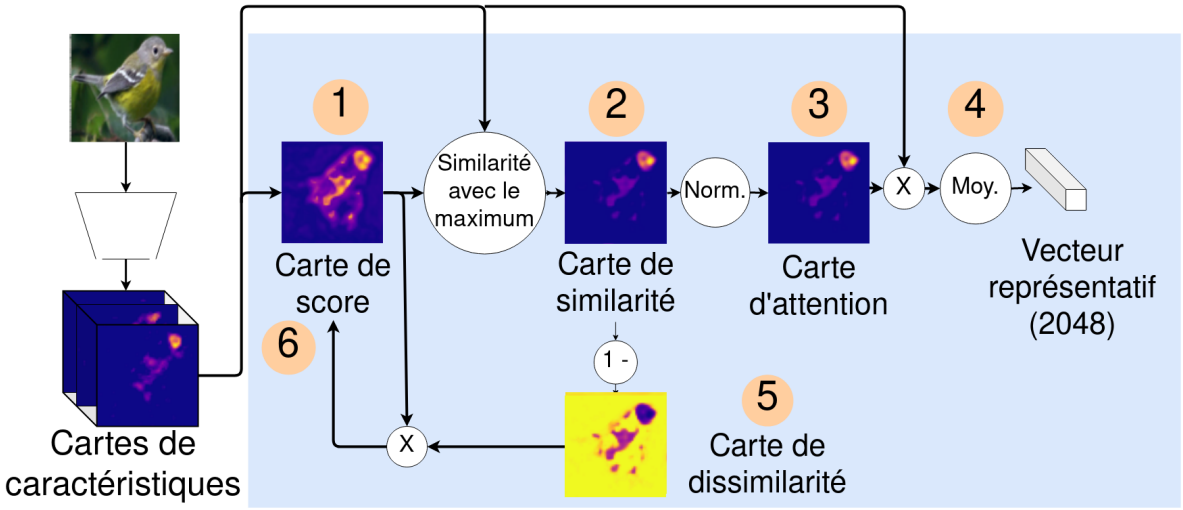


FIGURE 3.2 – Illustration de la méthode utilisée pour grouper les caractéristiques sans module neuronal dédié.

Algorithm 1 Identification des vecteurs représentatifs $\{\hat{f}_k\}$

Entrée : vecteurs de caractéristiques $\{f_i\}$

(1) $a_i \leftarrow ||f_i||_2$
pour $k = 1$ à N **faire**
  $i_{max} \leftarrow \underset{i}{\text{argmax}}\ a_i$
  pour tout i **faire**
    (2) $s_i \leftarrow \cos(f_i; f_{i_{max}})$
    (3) $w_i \leftarrow s_i / \sum_{i'} s'_i$
    (4) $\hat{f}_k \leftarrow \sum_i w_i \times f_i$
    (5,6) $a_i \leftarrow (1 - w_i) \times a_i$
  fin pour
fin pour
Retourne $\hat{f}_1, \hat{f}_2, ..., \hat{f}_N$

L'algorithme de regroupement des traits proposé est détaillé par l'Alg. 1 et illustré en Fig. 3.2. En bref, il sélectionne les N premiers vecteurs de caractéristiques singuliers (c'est-à-dire non similaires les uns aux autres) actifs dans le volume de caractéristiques et les affine en les regroupant avec des vecteurs similaires. Cet algorithme produit une matrice de caractéristiques $N \times M$, où M est la dimension des vecteurs. Pour simplifier, les vecteurs de caractéristiques sont indexés avec seulement $i \in [0, H \cdot W - 1]$, où $H \times W$ est la résolution des cartes de caractéristiques. Le i-ème vecteur est donc désigné par f_i.

Lors de la première itération, à chaque f_i est assigné un score d'activation a_i égal à sa norme euclidienne (étape 1 dans Alg. 1 et Fig. 3.2). Ensuite, le vecteur ayant la norme la plus élevée est sélectionné (étape 2). Cette procédure est motivée par le fait que, parmi tous les vecteurs de caractéristiques du volume de caractéristiques d'un CNN, ceux qui ont les normes les plus élevées ont un impact plus important sur le vecteur de caractéristiques final obtenu et contiennent des informations plus pertinentes pour la classification. Ainsi, en sélectionnant le vecteur à norme élevée, nous cherchons à reproduire ce phénomène dans notre architecture. Cette intuition est vérifiée empiriquement dans la section 3.2.3, où les résultats expérimentaux montrent que la précision diminue en choisissant des vecteurs aléatoirement.

Après avoir obtenu le vecteur caractéristique avec la norme maximale $f_{i_{\text{max}}}$, il est ensuite affiné avec d'autres vecteurs qui lui sont similaires (étape 2 à 4). Concrètement, la similarité cosinus s_i entre chaque caractéristique f_i et $f_{i_{\text{max}}}$ est calculée (étape 2). Ensuite, les similarités $\{s_i\}$ sont normalisées pour obtenir des poids $\{w_i\}$ tels que $\sum_i w_i = 1$ (étape 3). Le vecteur représentatif $\hat{f}_1$ est ensuite obtenu en calculant la moyenne pondérée de tous les vecteurs (étape 4). Ensuite, une carte de dissimilarité $1 - w_i$ est calculée (étape 5) et les scores d'activation $\{a_i\}$ sont mis à jour en les multipliant par les $1 - w_i$ (étape 6). Ce faisant, on peut s'assurer que le nouvel emplacement choisi i_{max} et son estimation $f_{i_{\text{max}}}$ sont différents de ceux choisis lors des itérations précédentes, tout en continuant à extraire des vecteurs à norme élevée. La procédure susmentionnée (lignes 3 à 8) est répétée encore N-1 fois pour obtenir au total N vecteur de caractéristique. Enfin, les vecteurs obtenus sont concaténés et sont passés à la couche de classification finale. Les poids $\{w_i\}$ sont vus comme des cartes d'attention pour mettre en évidence les parties locales qui contribuent le plus à chaque vecteur et donc à la décision finale, le rang des vecteurs indiquant le niveau d'importance des parties d'objet respectives. Il faut noter que, comme toutes les opérations sont faites seulement à partir du volume de caractéristiques, ce modèle d'attention est non paramétrique. Ce schéma peut être facilement intégré dans la plupart des réseaux neuronaux de classification profond, car ils intègrent souvent une couche d'agrégation qui peut être remplacée par la couche d'attention de BR-NPA.

3.1.3 Discussion

Après l'étape 4, nous pourrions répéter la procédure en sélectionnant simplement le vecteur ayant la deuxième norme la plus élevée pour construire un nouveau vecteur représentatif. Cependant, le deuxième vecteur de norme la plus élevée est susceptible d'être spatialement proche du vecteur de norme la plus élevée, c'est-à-dire similaire au vecteur de norme la plus élevée. Cela implique que la deuxième carte de similarité sera probablement similaire à la première et que le deuxième vecteur représentatif ne contiendra que peu ou pas d'informations supplémentaires par rapport au premier. Pour éviter cela, nous calculons une carte de dissimilarité $1 - w_i$ en inversant les poids (étape 5) et actualisons les scores d'activation $\{a_i\}$ en les multipliant par $1 - w_i$ pour tous les i (étape 6). Les informations fournies par la carte de dissimilarité permettent de sélectionner un vecteur qui n'est pas similaire au précédent. Cette multiplication permet donc de prendre en compte à la fois le critère de norme élevée et le critère de dissimilarité. En conséquence, les vecteurs dissimilaires à norme élevée seront choisis avant les vecteurs dissimilaires à norme faible. En d'autres termes, pour deux vecteurs qui ont la même w_i, celui qui a la norme la plus élevée sera choisi. Nous montrons dans section 3.2.3 que les 2ème et 3ème vecteurs représentatifs portent effectivement des informations à la fois saillantes et différentes de celles du 1er vecteur.

Il faut noter que plus le nombre de cartes d'attention N est grand, plus la visualisation est complexe. De plus, nous avons observé que l'utilisation de $N > 3$ n'améliore pas la précision de manière significative et rend la visualisation plus complexe. Pour ces raisons, nous utilisons $N = 3$ cartes d'attention par défaut dans ce travail. Le choix de N est discuté plus en détail en section 3.2.7.

La dimension du vecteur obtenu après la concaténation peut devenir grande lorsque N augmente ce qui pourrait nuire à l'entraînement du modèle. Cependant, nous utilisons de petites valeurs de N pour que le modèle reste interprétable. De plus, nous observons dans notre analyse de l'impact de N (figure 3.10) que même avec $N = 64$ et un vecteur de caractéristiques de taille 131072, la précision est similaire à celle obtenue en utilisant N=3 et un vecteur de taille $2048 \times 3 = 6144$. Cela pourrait indiquer que l'augmentation de la taille du vecteur n'entrave pas l'optimisation. Une autre explication est que les objets peuvent n'avoir en moyenne que 2 ou 3 parties discriminantes, empêchant les vecteurs représentatifs 4 à 64 de contenir des informations pertinentes, les rendant ainsi ignorés par la couche de classification lors de l'optimisation. Une inspection des poids donnés aux caractéristiques de ces vecteurs confirmerait cette hypothèse mais nous laissons cela pour un travail futur.

3.2 Expériences

Le modèle proposé a d'abord été évalué dans le cadre de diverses tâches de vision par ordinateur, notamment la classification d'images à grain fin [152, 101, 84], la classification avec peu d'exemples [120] et la réidentification de personnes [174]. En outre, des études approfondies d'ablation ont été menées pour valider les stratégies choisies.

Nous avons comparé le modèle d'attention BR-NPA proposé avec des modèles de la littérature comme B-CNN [93], ProtoPNet [21], ProtoTree [107], IBN [67], RGA [171], HA-CNN [89], OS [177] et l'attention croisée (CA) [62], ainsi que des méthodes post-hoc telles que Grad-CAM [127], Grad-CAM++ [19], Guided backpropagation [136], Score-CAM [156], RISE [115], VarGrad [2], SmoothGrad [135], et une méthode post-hoc de base que nous proposons : "Activation Map" (AM) ("carte d'activation"), qui consiste simplement à visualiser la norme euclidienne de chaque vecteur du volume de caractéristiques.

Les modèles d'attention ont été entraînés à l'aide du code des auteurs et nous avons utilisé Captum [83] pour les méthodes de visualisation post-hoc. Pour la classification fine, nous rapportons également, à titre de référence, la performance de l'attention à deux niveaux (ADN) [161], MG-CNN [153], FCAN [94], ST-CNN [73] et MA-CNN [172]. Ces modèles n'ont pas été inclus dans les évaluations qualitatives ou quantitatives des cartes de saillance car il s'agit de modèles d'attention dure qui produisent des boîtes englobantes, ce qui empêche une comparaison avec les cartes d'attention douces produites par le BR-NPA, mais il s'agit tout de même de modèles d'attention avec lesquels il est intéressant de comparer BR-NPA.

Le modèle B-CNN est une architecture de base simple qui peut être appliquée directement à un grand nombre de tâches, ce qui le rend approprié pour être appliqué à d'autres tâches que celle pour laquelle il a été développé. Par conséquent, parmi les modèles d'attention mentionnés ci-dessus, seul le B-CNN a été entraîné sur toutes les tâches considérées, puisque les autres modèles d'attention ont été développés/conçus pour une certaine tâche en particulier. D'autre part, les algorithmes post-hoc étant des méthodes génériques, chacun d'entre eux a été appliqué à l'ensemble des trois tâches. Les algorithmes post-hoc ont été appliqué sur un CNN ordinaire sans attention entraîné avec les mêmes paramètres que les modèles d'attention.

3.2.1 Détails d'implémentation

Comme mentionné dans la section 3.1.2, nous avons choisi $N = 3$ pour BR-NPA car il s'agit d'un bon compromis entre précision et interprétabilité. Plus de détails sur la sélection de N peuvent être trouvés en section 3.2.7. Comme le nombre de cartes d'attention générées pour une

image varie considérablement d'un modèle à l'autre (*e.g.*, 32 pour InterByParts, jusqu'à 2000 pour ProtoPNet), nous avons sélectionné 3 cartes d'attention parmi celles produites par chaque modèle. Cela garantit une comparaison qualitative concise et équitable entre les modèles. Seul le modèle B-CNN a été entraîné en utilisant 3 parties, car l'utilisation d'un plus grand nombre de cartes n'améliore pas la précision et demande plus de mémoire vidéo pendant l'entraînement. Pour les autres modèles, nous avons conservé le même nombre de cartes d'attention que celui indiqué dans les articles originaux. Pour sélectionner les cartes d'attention, nous avons adopté différentes stratégies. Pour ProtoTree et ProtoPNet, nous avons choisi les cartes les plus actives, c'est-à-dire celles qui ont le score de similarité le plus élevé avec l'image d'entrée, car elles sont les plus pertinentes pour la tâche cible. Pour RGA et HA-CNN, nous avons utilisé les cartes d'attention générées par les 3 dernières couches, car elles sont les plus proches des couches décisionnelles. Une fois que 3 cartes sont sélectionnées, elles sont visualisées avec une image RVB où la première, deuxième et troisième carte sont respectivement représentées en rouge, vert et bleu. Les trois cartes sont également multipliées par la norme des vecteurs de caractéristiques correspondant, afin de rendre plus visible les vecteurs qui ont eu une grande importance durant la phase d'agrégation. Le modèle OS et les algorithmes de saillance ne génèrent qu'une seule carte d'attention, elles ont été visualisées à l'aide d'une simple carte de chaleur.

Pour toutes les expériences et tâches, les hyper-paramètres ont été estimés à l'aide de l'estimateur de Parzen arborescent de la librairie Optuna [5], sauf dans la section 3.2.3, où nous avons utilisé les valeurs d'hyper-paramètres suivantes : un taux d'apprentissage de 0.001, l'optimiseur SGD avec un momentum de 0.9 et une taille de lot de 12. Les détails d'adaptation de l'attention proposée pour les différentes tâches sont donnés ci-dessous.

Classification d'images à grain fin. Pour BR-NPA et le modèle de base B-CNN, le réseau ResNet-50 [58] pré-entraîné sur le jeu de données ImageNet [122] a été considéré comme le réseau de base car il s'agit de l'une des architectures de réseau les plus couramment utilisées dans divers domaines. Pour obtenir des cartes de caractéristiques à plus haute résolution, le pas des blocs de sous-échantillonnage dans les couches 3 et 4 du réseau ResNet-50 a été fixé à 1 au lieu de 2, ce qui a fait passer les cartes de caractéristiques d'une taille de 14×14 à 56×56. Le modèle professeur utilisé pour la distillation est aussi un modèle BR-NPA mais avec une résolution de 14×14.

Classification d'image avec peu d'exemples. Le modèle E3BM proposé par [96] est l'un des meilleurs modèles de classification avec peu d'exemple de l'état de l'art. Il a donc été

utilisé comme architecture, et combiné avec B-CNN, CA et BR-NPA. Pour comparer BR-NPA à B-CNN et CA, la couche d'agrégation par moyenne du modèle E3BM a été remplacée par les couches d'attention susmentionnées.

Pour B-CNN et BR-NPA, la couche de classification originale a également été étendue pour prendre en entrée la concaténation des 3 vecteurs de caractéristiques produit par la couche d'attention. Une telle modification n'a pas été nécessaire pour CA car elle n'étend pas la taille du vecteur de caractéristiques produit. Le modèle a ensuite été entraîné comme décrit par [96] sur une des base de données de classification avec peu d'exemple parmi les plus populaires, TieredImagenet [120], en utilisant des poids pré-entraînés fournis par les auteurs pour réduire la durée d'entraînement. Nous utilisons le même modèle que [96], à savoir ResNet-12. Le pas des blocs de sous-échantillonnage dans les couches 2 et 3 du ResNet-12 a été fixé à 1 au lieu de 2. En conséquence, la résolution des cartes de caractéristiques a été portée de 5×5 à 21×21. Le modèle professeur utilisé pour la distillation est le modèle E3BM pré-entraîné non-modifié dont les poids ont été fournis par les auteurs.

Réidentification de personne. De manière similaire à la classification avec peu d'exemples, l'un des modèles de réidentification les plus performants a été utilisé, à savoir DG-Net [175], et a été adapté avec les différents modules d'attention étudiées. Pour BR-NPA et B-CNN, la couche d'agrégation par moyenne dans le module d'identification utilisé par [175] a été supprimée et remplacée par la couche d'attention, et la couche linéaire finale a également été étendue pour correspondre à la nouvelle taille du vecteur de caractéristiques. La couche de classification a aussi dû être étendue le long de l'axe des caractéristiques afin de pouvoir recevoir la concaténation des 3 vecteurs et donc passer d'une taille de 512×2048 à $512 \times 2048 \cdot N$. Cette procédure a été utilisée pour entraîner le DG-Net en combinaison avec le BR-NPA et le B-CNN. Ensuite, le modèle a été entraîné comme expliqué dans [175] sur le jeu de données Market-1501 [174]. Il faut souligner que le modèle DG-Net a une architecture complexe qui est difficile à entraîner. Pour cette raison, nous avons donc utilisé les poids pré-entraînés fournis par les auteurs et modifié l'architecture du DG-Net pour faciliter l'entraînement.

Nous utilisons la même colonne vertébrale que DG-Net, à savoir ResNet-50, sauf pour RGA, HA-CNN et OS, où le modèle ResNet-50 a été remplacé par le CNN proposé par ces méthodes, entraîné dans la même configuration que DG-Net. Les pas des couches 2 et 3 du ResNet-50 ont été fixés à 1 au lieu de 2, sachant que le pas de la couche 4 était déjà fixé à 1 par les auteurs du modèle. Cela a permis d'augmenter la résolution des cartes de caractéristiques de 16×8 à 64×32. La méthode proposée par les auteurs de DG-Net propose déjà de distiller un modèle

professeur dans le modèle entraîné, nous avons donc repris la même procédure de distillation.

3.2.2 Performance et facilité d'intégration

Classification d'images à grain fin. Les performances du modèle proposé et des modèles d'attention de l'état de l'art en termes de précision sur trois ensembles de données à grain fin, à savoir CUB-200-2011 [152], FGVC-Aircraft [101] et Standford cars [84] sont présentées dans le tableau 3.1. On observe que BR-NPA surpasse les autres méthodes orientées vers l'interprétabilité sur les trois ensembles de données.

Modèle	Attention	CUB	FGVC	Stanford cars
ADN [161]	Dure	77.9	-	-
MG-CNN [153]	Dure	81.7	-	-
FCAN [94]	Dure	82.0	-	-
ST-CNN [73]	Dure	84.1	-	-
MA-CNN [172]	Dure	85.4	88.4	91.7
InterByParts [68]	Douce	81.9	87.9	89.5
ProtoTree [107]	Douce	82.1	81.3	85.7
B-CNN [93]	Douce	84.1	84.2	91.3
B-CNN (notre impl.)	Douce	82.9	88.6	89.9
ProtoPNet [21]	Douce	84.8	84.0	85.8
BR-NPA	Douce	**85.5**	**89.6**	**92.2**

TABLE 3.1 – Performance sur les jeux de données de classification fine.

Des visualisations des cartes d'attention sont présentées dans la figure 3.3. Parmi toutes les images, les cartes d'attention produites par BR-NPA mettent en évidence des détails plus précis des objets. Par exemple, dans les rangées 4-6, les régions mises en évidence sont la plupart du temps les queues ou les ailes des avions, ce qui permet de différencier les différents modèles d'avions. De même, les autres modèles d'attention sont également capables d'identifier des parties locales importantes similaires, mais ils sont généralement moins précis et cohérents. À l'inverse, les régions mises en évidence par les approches Grad-CAM, Grad-CAM++, AM, RISE et Score-CAM sont nettement plus grossières et couvrent l'ensemble des objets, tandis que celles des approches Guided-Backpropagation, VarGrad et SmoothGrad sont extrêmement éparses, ce qui rend difficile l'identification des zones importantes et les répartit parfois de manière aléatoire sur l'ensemble des images. Aucune de ces régions d'attention ne fournit d'informations précises qui aident à la prise de décision d'une classification à grain fin.

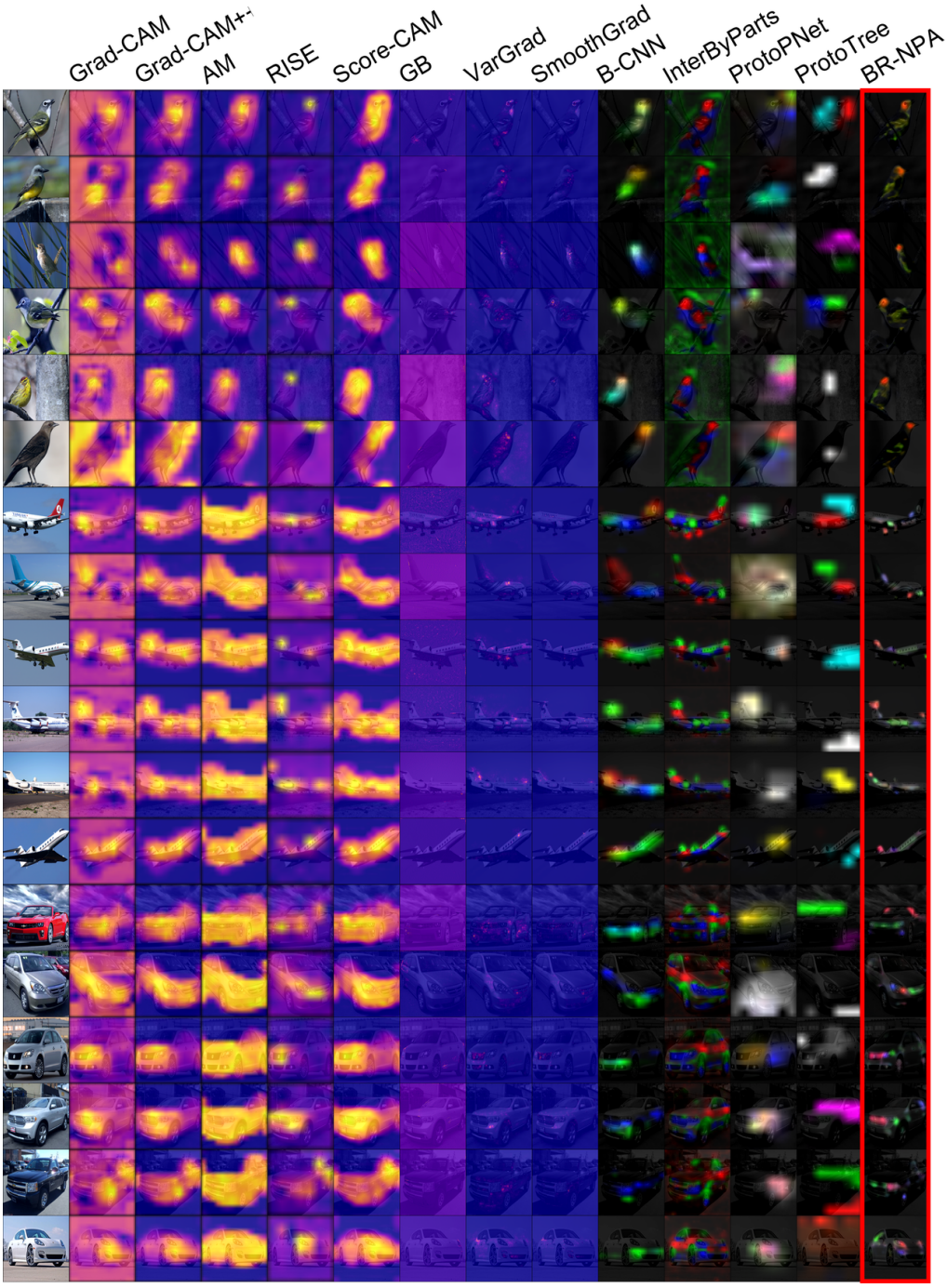


FIGURE 3.3 – Comparaison de différentes explications visuelles avec BR-NPA pour la classification fine sur 3 jeux de données : CUB-200-2011 lignes 1-6, FGVC-Aircraft lignes 7-12, et les voitures de Standford lignes 13-18. BR-NPA produit des cartes d'attention détaillées centrées sur les parties sémantiques de l'objet, alors que les autres méthodes produisent soit des cartes floues couvrant l'ensemble de l'objet, soit des cartes extrêmement éparses rendant difficile l'identification des zones importantes.

Re-identification de personne. La précision du DG-Net équipé de différentes méthodes d'attention sur le jeu de données Market-1501 [174] est indiqué dans le tableau 3.2. Par rapport aux autres méthodes d'attention, la version à basse résolution de BR-NPA obtient les meilleures performances. Les performances sont très proches de celle du modèle DG-Net original, ce qui montre la facilité avec laquelle il est possible d'intégrer BR-NPA dans une architecture existante déjà entraînée.

Modèle	Réference Précision	Attention	Résolution	Précision
DG-Net	94.8	B-CNN	16×8	80.1
		HA-CNN	10×4	86.6
		OS	16×8	91.2
		RGA	16×8	93.0
		BR-NPA	16×8	**93.6**
			64×16	88.1

TABLE 3.2 – Performance pour la tâche de réidentification des personnes.

Des exemples de cartes d'attention obtenues sont présentées en figure 3.4. Comme pour la classification fine, les cartes d'attention produites par BR-NPA mettent en évidence les détails locaux importants des vêtements des personnes, qui sont essentiels pour la réidentification, compte tenu des variations intra-classes significatives entre les différentes caméras. Par exemple, les régions mises en évidence comprennent les chaussures (lignes 2,3,5 et 6), les bretelles d'un sac à dos (ligne 8 et 9) ou le cou de la personne (lignes 1,3,6 et 7) qui aident à différencier les différentes personnes. Les méthodes B-CNN et OS ne se concentrent que sur des parties plus larges de la personne, par exemple le haut du corps (lignes 1 et lignes 3 à 9), et les méthodes RGA et HA-CNN se concentrent à peine sur le corps de la personne. De même que pour la reconnaissance fine, les régions mises en évidence par les approches Grad-CAM, Grad-CAM++, AM, RISE et Score-CAM couvrent généralement la totalité des objets, tandis que celles des méthodes Guided-Backpropagation, VarGrad et SmoothGrad sont extrêmement éparses, ce qui rend difficile l'identification des zones importantes.

Il est démontré ici que BR-NPA pourrait être facilement intégré dans une architecture complexe pré-entraînée sans souffrir d'une perte significative de précision tout en produisant une visualisation interprétable.

Classification avec peu d'exemples. Nous avons testé la combinaison du modèle E3BM avec 3 modèles d'attention : B-CNN, CA et les modules BR-NPA. Chaque combinaison a été évaluée

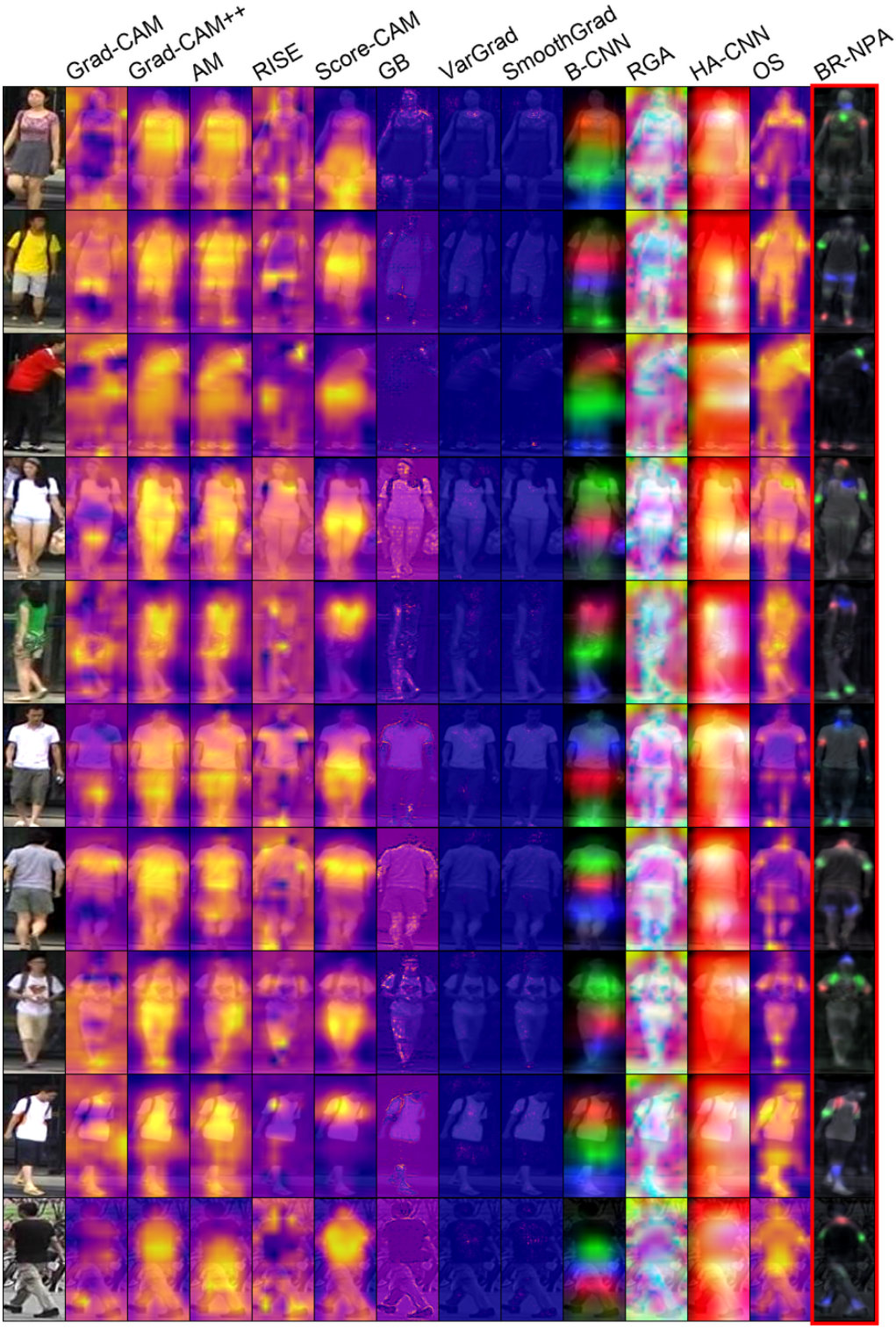


FIGURE 3.4 – Comparaison de différentes explications visuelles avec BR-NPA pour la tâche de réidentification sur le jeu de données Market-1501.

avec 5 classes et 5 exemples par classe sur le jeu de données CIFAR-FS [14]. Les précisions moyennes sont indiquées dans le tableau 3.3. Nous observons que l'intégration de la version de BR-NPA en résolution 5×5 améliore la précision du modèle E3BM original, et les versions 5×5 et 21×21 de BR-NPA offrent des performances supérieures à B-CNN et CA.

Cela montre que BR-NPA est un module polyvalent qui peut être intégré dans une architecture pré-entraînée existante et maintenir les performances du modèle.

Modèle	Référence Précision	Attention	Résolution	Précision
E3BM	85.8	CA	5×5	81.9
		B-CNN	5×5	83.2
		BR-NPA	5×5	**85.9**
			21×21	85.3

TABLE 3.3 – Performance pour la tâche de classification avec peu d'images.

Les cartes d'attention sont visibles en figure 3.5. D'une part, contrairement à la classification fine et à la réidentification de personnes, dans la tâche de classification avec peu d'images, les cartes d'attention générées par BR-NPA mettent en évidence des zones légèrement plus grandes de l'objet, tandis que la plupart des parties locales sont encore bien déterminées et peuvent être distinguées les unes des autres. Par exemple, BR-NPA sépare le corps de la tête (lignes 1-3) ou les oreilles du nez (lignes 7-9), qui sont tous des composants clés pertinents pour la reconnaissance des catégories correspondantes. Ces observations démontrent la faisabilité de l'adaptation de BR-NPA pour la tâche de classification avec peu d'exemples. D'autre part, le B-CNN n'est pas capable de différencier les différentes parties locales car il ne génère qu'une seule carte d'attention significative (en vert), l'autre (en rouge) présentant un comportement apparemment dégénéré. De même, CA ne différencie pas non plus les parties de l'objet et présente un comportement encore plus dégénéré. Encore une fois, les régions mises en évidence par les approches Grad-CAM, Grad-CAM++, AM, RISE et Score-CAM couvrent généralement la totalité des objets, tandis que celles de Guided-Backpropagation, VarGrad et SmoothGrad sont extrêmement clairsemées, ce qui rend difficile l'identification des zones importantes et les répartit parfois de manière aléatoire sur toute l'image.

Comparaisons des cartes d'attention obtenues à partir de différentes tâches. Les cartes d'attention obtenues à partir de différentes tâches mettent en évidence l'objet cible à différents niveaux de granularité. Pour les tâches de classification fine et de réidentification de personnes, BR-NPA produit des cartes d'attention plus précises qui se concentrent sur des parties plus fines

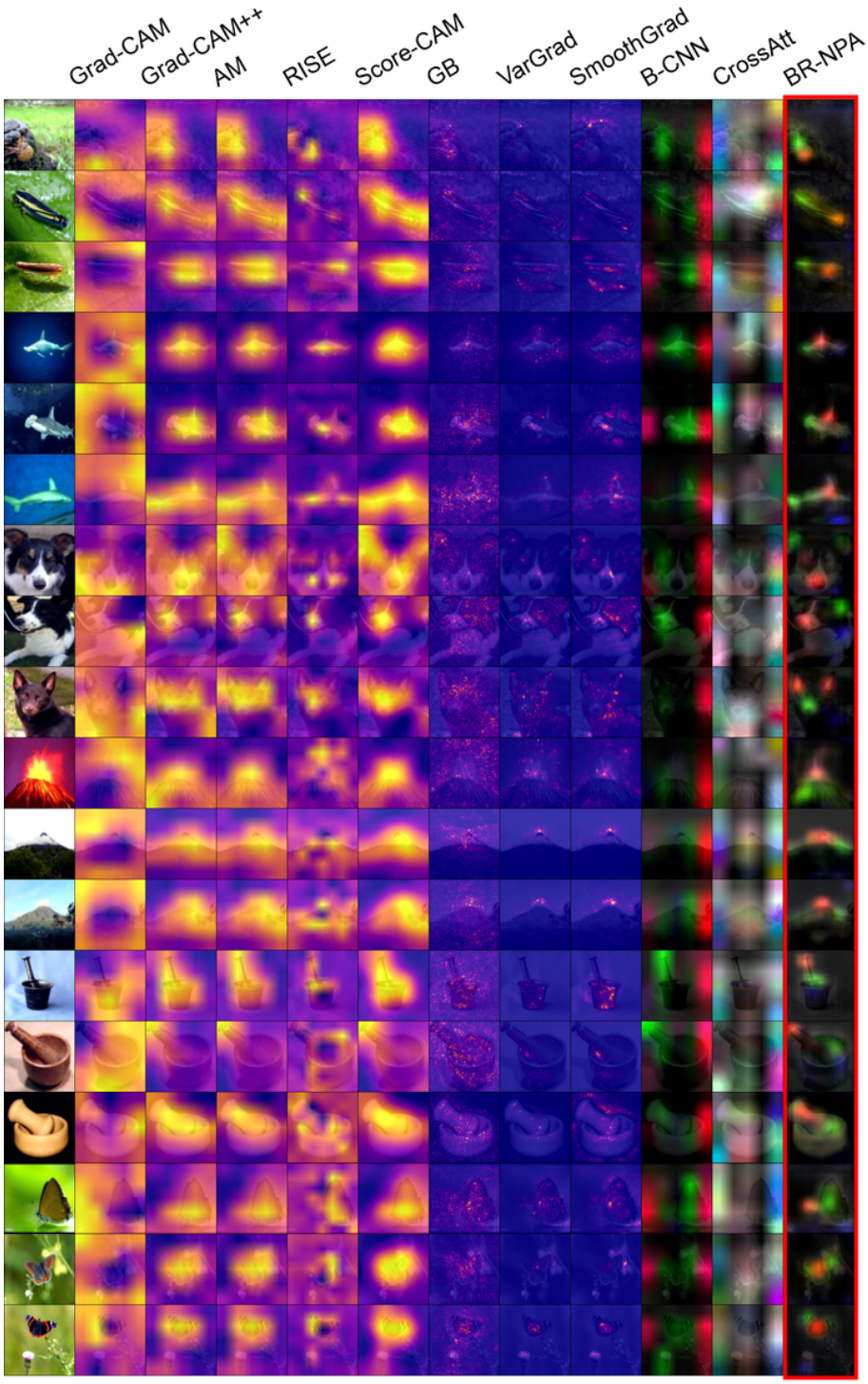


FIGURE 3.5 – Comparaison de différentes explications visuelles avec BR-NPA pour la tâche de classification avec peu d'images sur le jeu de données TieredImagenet.

de l'objet. En revanche, pour la classification avec peu d'exemples, le modèle proposé tend à se concentrer sur l'objet entier, car l'objectif de ces images est de distinguer des catégories plutôt que des sous-catégories. Cela montre l'adaptabilité du modèle proposé aux différentes tâches.

3.2.3 Étude ablative

Impact du critère de sélection des vecteurs et de la stratégie de raffinement. Trois modèles ont été entraînés pour explorer l'impact du critère de sélection des vecteurs basé sur leur norme euclidienne et l'impact de la stratégie de raffinement des vecteurs, à savoir :

1. un modèle où les vecteurs de caractéristiques sont sélectionnés de façon uniformément aléatoire parmi tous les vecteurs disponibles au lieu de sélectionner les vecteurs actifs ;
2. un modèle ablatif sans phase de raffinement, qui concatène simplement tous les vecteurs actifs singuliers et les transmet directement à la couche de classification ;
3. un modèle avec sélection aléatoire et sans phase de raffinement.

Au cours de cette expérience, les 3 modèles ainsi que le modèle original ont été entraînés avec des pas des couches de sous-échantillonnage des couches 3 et 4 réduits de 2 à 1 afin d'augmenter la résolution de 14×14 à 56×56 (cf. section 3.2.1) sans distillation ni optimisation des hyper-paramètres. Les résultats présentés dans le tableau 3.4 sur le jeu de données CUB-200-2011 montre l'intérêt pour la précision du modèle de sélectionner les vecteurs actifs et d'inclure la phase de raffinement dans le calcul des vecteurs.

Critère de sélection des vecteurs	Raffinement	Précision
Aléatoire	Non	0.5
Aléatoire	Oui	0.5
Actif	Non	71.2
Actif	Oui	**80.3**

TABLE 3.4 – Résultat de l'étude ablative sur le jeu de données CUB-200-2011.

Étude du pouvoir de discriminabilité des vecteurs représentatifs. Afin d'étudier le pouvoir discriminant des vecteurs extraits en fonction de leur rang, 5 couches de classification ont été ajoutées à BR-NPA et ont reçu en entrée un ensemble différent de vecteurs représentatifs. Plus précisément, au lieu de recevoir les 3 vecteurs représentatifs, *c'est-à-dire*, $\{\hat{f}_1, \hat{f}_2, \hat{f}_3\}$, les 5 couches ne reçoivent respectivement que le vecteur ou la combinaison de vecteurs suivants : $\{\hat{f}_1, \hat{f}_2\}$, $\{\hat{f}_2, \hat{f}_3\}$, $\{\hat{f}_1\}$,$\{\hat{f}_2\}$ et $\{\hat{f}_3\}$. Chaque couche supplémentaire a été entraînée en ajoutant

un terme d'entropie croisée à la fonction de perte. La somme est ensuite divisée par le nombre total de termes, soit 6, de sorte que la nouvelle fonction de perte finale est la moyenne des 6 termes d'entropie croisée. Pour empêcher les couches supplémentaires de modifier les vecteurs de caractéristiques, nous empêchons leurs gradients de se propager vers les cartes de caractéristiques.

Comme pour l'étude précédente, le modèle avec les couches supplémentaires a été entraîné en résolution 56×56 sans distillation. La précision de chacune des couches supplémentaires (et de la couche principale) est résumée dans le tableau 3.5 sur le jeu de données CUB-200-2011. Comme on peut le voir, lorsqu'on utilise un seul vecteur, la performance diminue avec le rang des vecteurs représentatifs (de 1 à 3). Lorsque l'on retire un des vecteurs de l'ensemble complet, la précision diminue également. Cette observation prouve indirectement que la contribution de chaque carte d'attention à la tâche est bien liée à son rang. Notez que les performances du modèle utilisant la méthode de sélection active avec affinage à partir de l'étude d'ablation précédente (80.3%) sont différentes des performances du modèle utilisant les trois vecteurs $\{\hat{f}_1, \hat{f}_2, \hat{f}_3\}$. Cela est dû au fait que, dans cette dernière, la perte est une moyenne de 6 termes, ce qui signifie que le terme d'entropie croisée du modèle est multiplié par un facteur $1/6$, ce qui équivaut à utiliser un taux d'apprentissage plus petit d'un facteur 6 par rapport à l'expérience précédente.

Vecteur(s)	$\{\hat{f}_1, \hat{f}_2, \hat{f}_3\}$	$\{\hat{f}_1, \hat{f}_2\}$	$\{\hat{f}_2, \hat{f}_3\}$	$\{\hat{f}_1\}$	$\{\hat{f}_2\}$	$\{\hat{f}_3\}$
Acc.	**82.8**	82.7	81.4	81.2	80.0	66.3

TABLE 3.5 – Impact du nombre et du rang des vecteurs représentatifs sur la précision, en utilisant le jeu de données CUB-200-2011.

3.2.4 Temps d'entraînement

Pour évaluer le rapport bénéfice/coût de l'utilisation de 3 vecteurs au lieu de 2 ou 1, nous avons calculé le temps moyen d'entraînement sur 20 époques avec le méta-paramètre BR-NPA par défaut $N = 3$, mais aussi avec $N = 2$ et $N = 1$. Les modèles ont également été entraînés en haute résolution sans distillation ni optimisation des hyper-paramètres. Nous avons obtenu respectivement une durée moyenne des époques d'entraînement de 498.88, 496.03, et 480.60 secondes pour $N = 3$, 2, et 1, avec des écarts types de 0.65, 1.12, et 1.06. Cela montre que l'augmentation du temps de calcul lors de l'utilisation de 3 vecteurs au lieu de 2 ou 1 est limitée car la plupart du temps de calcul est consacré aux cartes de caractéristiques produites par ResNet-50. Par conséquent, l'utilisation de 3 vecteurs au lieu de 2 ou 1 permet d'extraire plus d'informations significatives à partir des mêmes cartes de caractéristiques avec une augmentation

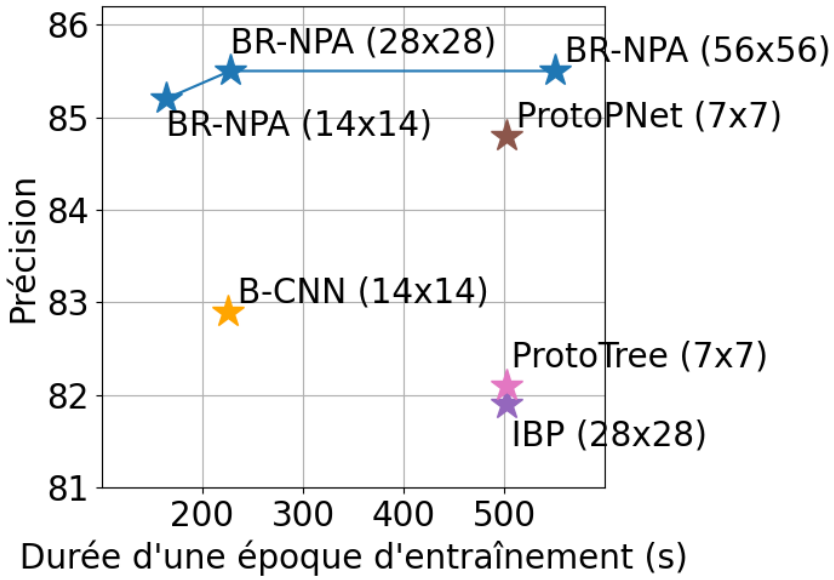


FIGURE 3.6 – Précision sur le jeu de test en fonction de la durée moyenne d'une époque. Chaque point est annoté avec le nom du modèle et la résolution correspondante.

du coût de calcul minimale.

Efficacité de BR-NPA en fonction de la résolution. Nous étudions maintenant plus en détail le temps d'entraînement de BR-NPA en le comparant à celui d'autres modèles. Pour simplifier, nous utilisons les mêmes modèles que pour la classification à grain fin, c'est-à-dire ProtoPNet, ProtoTree et IBP. Nous mesurons également le temps d'apprentissage d'un BR-NPA basse résolution (14×14) et d'un BR-NPA moyenne résolution (28×28) avec distillation où seul le pas de la couche 4 est fixé à 1 et le pas de la couche 3 est laissé à la valeur par défaut de 2. Comme dans le paragraphe précédent, nous avons calculé le temps d'entraînement moyen sur 20 époques. Nous visualisons les résultats en figure 3.6 en fonction de la précision obtenue pendant la phase de test. Cette figure montre que BR-NPA est plus efficace en termes de rapport précision/temps d'apprentissage que les autres modèles.

Efficacité de BR-NPA en fonction du nombre de couches. Nous avons également comparé le temps d'exécution en entraînement et l'empreinte mémoire de BR-NPA avec B-CNN. Tout d'abord, nous avons calculé le temps nécessaire au traitement d'un lot de 10 images à une résolution de 448×448 pendant l'entraînement pour plusieurs réseaux : ResNet-18, ResNet-34, ResNet-50, ResNet-101 et ResNet-152 en haute (56×56) et basse (14×14) résolutions.

La figure 3.7 rapporte le temps d'exécution en secondes observé sur 100 lots. Les résultats montrent que BR-NPA est plus rapide que B-CNN pour un même niveau de précision. Lors de l'entraînement avec ResNet-50, nous avons également calculé la taille du plus grand lot qui peut être utilisé pendant l'entraînement de BR-NPA et B-CNN avec sur 2 GPU de 16 Go. La figure 3.7 montre que BR-NPA consomme moins de mémoire GPU que B-CNN pour une même résolution et un même niveau de précision.

3.2.5 Étude de l'impact de la haute-résolution

Les effets de l'utilisation d'une plus grande résolution de carte de caractéristiques sur les performances d'un CNN, d'un B-CNN [92] et d'un BR-NPA sont explorés ici, sur le jeu de données CUB-200-2011. Nous comparons deux résolutions, à savoir 14×14 et 56×56. La première est la résolution de sortie par défaut et la seconde a été obtenue en réduisant les pas des couches 3 et 4 à de 2 à 1. Nous comparons aussi l'impact de la distillation sur les modèles en haute-résolution.

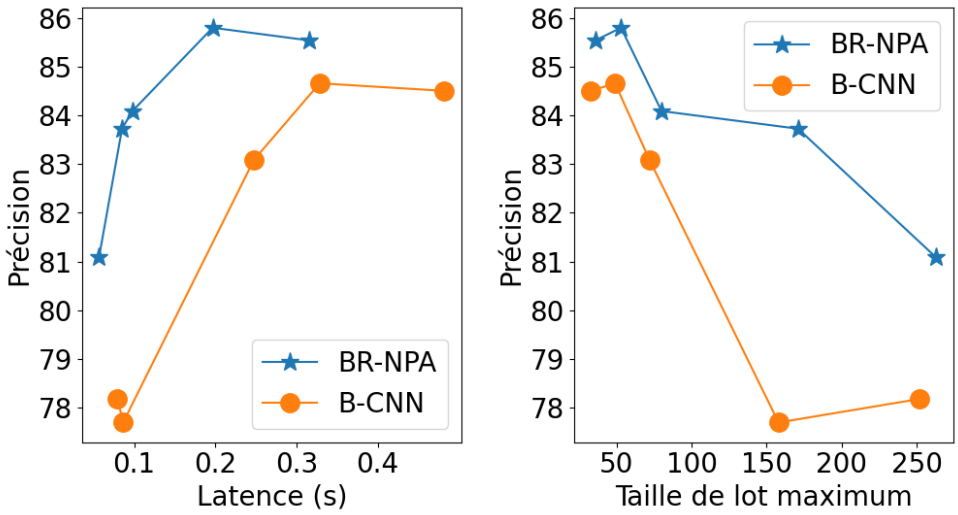
Idéalement, la carte d'attention devrait être clairsemée, c'est-à-dire se concentrer sur une petite zone locale de l'image pour qu'elle mette en valeur les sous-parties de l'objet qui sont pertinentes pour la classification fine. Pour quantifier cela, nous proposons d'utiliser la métrique de parcimonie introduite précédemment, définie comme suit :

$$\text{Parcimonie} = \frac{1}{S'_{mean}}, \tag{3.2}$$

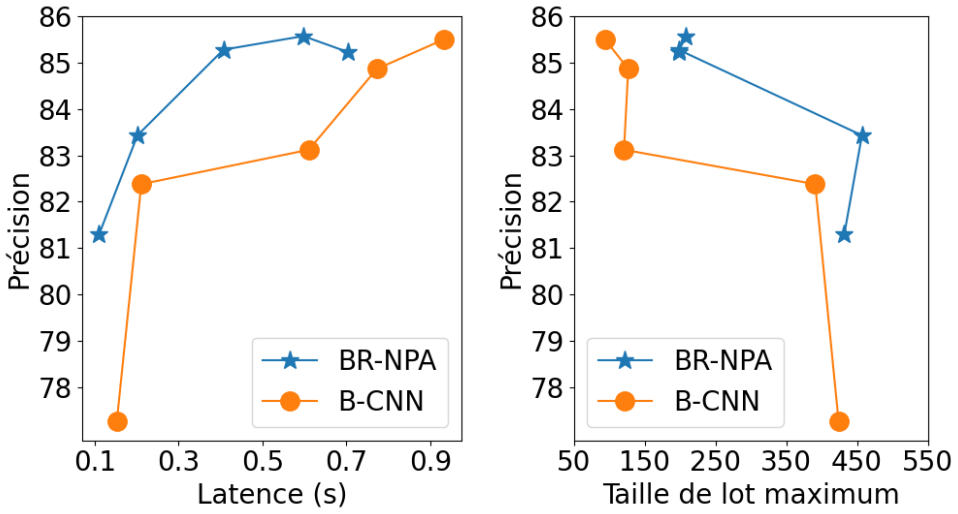
où S' est la carte de saillance S normalisée entre 0 et 1 comme suit : $S' = (S - S_{min})/(S_{max} - S_{min})$ et S'_{mean} est la valeur moyenne de S'.

Pour rappel, une carte d'attention focalisée sur une petite partie de l'image a une faible valeur moyenne et donc une valeur de parcimonie élevée. À l'inverse, une carte couvrant uniformément toute l'entrée a une faible parcimonie. Étant donné que B-CNN et BR-NPA génèrent plusieurs cartes par image, il est d'abord nécessaire de les agréger avant de pouvoir calculer la parcimonie. Pour cela, nous calculons une carte moyenne à partir des cartes multiples produites par le modèle.

Le tableau 3.6 montre les résultats de cette étude. Tout d'abord, les modèles haute résolution sans distillation ont des performances nettement plus faibles que celle de leur professeur pour le CNN et BR-NPA. Au contraire, les modèles haute résolution entraîné avec la distillation obtiennent des performances similaires à celle de leur professeur. De plus, les cartes de ces modèles ont une parcimonie plus importante que celle de leur professeur. Cela montre que la distillation combinée à une augmentation de la résolution des cartes de caractéristiques permet



(a) Basse résolution (14×14)



(b) Haute résolution (56×56)

FIGURE 3.7 – Comparaison de l'efficacité de B-CNN et BR-NPA pendant l'entraînement. À gauche : latence en haute et basse résolutions. À droite : taille maximale des lots en haute et basse résolution. Les mesures sont effectuées sur deux GPU avec une mémoire totale de 32 GB. La latence comprend l'inférence et la rétropropagation.

de générer des cartes d'attentions plus précises et localisée sans compromis sur la précision.

Modèle	Résolution des cartes de caractéristiques	Distillation	Précision	Parcimonie
CNN	14×14	-	84.2	9.52
	56×56	✗	79.2	21.27
	56×56	✓	83.1	38.46
B-CNN	14×14	-	82.9	11.23
	56×56	✗	82.4	37.03
	56×56	✓	84.6	19.60
BR-NPA	14×14	-	85.2	7.46
	56×56	✗	82.4	4.60
	56×56	✓	85.5	21.73

TABLE 3.6 – Impact de la résolution sur les performances en utilisant le jeu de données CUB-200-2011
.

De plus, nous avons également évalué l'impact de l'augmentation de la résolution des cartes de caractéristiques sur l'interprétabilité de la carte d'attention de manière qualitative. Nous avons entraîné un CNN élève à imiter un CNN professeur à plus faible résolution avec distillation et visualisons les cartes d'activation pour le CNN professeur et élève dans la figure 3.8. On peut observer que les cartes d'attention du CNN à plus haute résolution sont plus interprétables car elles sont plus clairsemées et les zones activées sont situées sur les sous-parties de l'objet au lieu de l'objet entier.

FIGURE 3.8 – De gauche à droite : l'image original, les cartes d'attention du CNN en basse résolution et les cartes du CNN en haute résolution.

3.2.6 Hiérarchie de discriminabilité des parties

La figure 3.9 montre que BR-NPA peut permettre de détecter s'il existe une hiérarchie *c'est-à-dire,* un classement du niveau d'importance entre les parties qui contribuent à la prise de décision de la tâche. Par exemple, dans le jeu de données sur les oiseaux, la première carte (en rouge) se concentre davantage sur la tête et la deuxième carte (en vert) sur le corps, ce qui indique que la tête est plus discriminante que le corps pour l'identification de l'espèce d'oiseau.

De plus, la carte bleue n'est pas visible sur le jeu de données des oiseaux ce qui implique qu'elle s'est concentrée sur les vecteurs avec des normes très faibles par rapport à ceux sélectionnés par la carte rouge ou verte. Les vecteurs de caractéristiques à faible norme euclidienne ont un impact négligeable sur la prédiction finale de la tâche. En d'autres termes, ces caractéristiques à faible norme ne sont pas pertinentes pour la décision. La visualisation sur le jeu de données d'avions montre que les trois cartes sont souvent visibles et que chacune d'entre elles peut se concentrer sur diverses parties comme la queue, les roues, les moteurs, etc. Cela indique que, selon le modèle, il est nécessaire de se concentrer sur trois parties car elles ont un niveau d'importance équivalent. De même, sur le jeu de données des voitures, la première et la deuxième carte se concentrent sur l'entrée d'air avant et les feux, tandis que les cartes bleues ont tendance à cibler d'autres détails comme le clignotant (colonne 2) ou les détails de la carrosserie (colonne 1). Les deux parties les plus importantes sont l'entrée d'air avant et les phares. Certains détails comme des parties de la carrosserie sont également discriminants dans une moindre mesure. On voit donc que BR-NPA facilite la découverte d'une hiérarchie parmi les parties de l'objet en fonction de leur pertinence pour la tâche.

3.2.7 Impact du nombre de cartes d'attention sur la précision

Le nombre de cartes d'attention N peut impacter les performances. Pour explorer cette influence, nous avons examiné différentes valeurs de N et évalué la précision correspondante pour B-CNN et BR-NPA pour la tâche de classification à grain fin. Les résultats sont présentés dans la figure 3.10. On observe qu'en faisant varier N les performances ne changent pas significativement pour les deux modèles. Par exemple, des valeurs de N égales à 16 ou 32 entraînent de faibles gains de précision par rapport à $N = 3$. Cependant, cela multiplie le nombre de cartes à visualiser, ce qui nuit à l'interprétabilité du modèle, sans que la précision ne s'améliore significativement. Étant donné que ce modèle est conçu pour être interprétable, nous avons choisi de n'utiliser que des cartes d'attention avec $N = 3$.

De plus, la figure 3.10 montre que la valeur de N qui maximise les performances de B-CNN

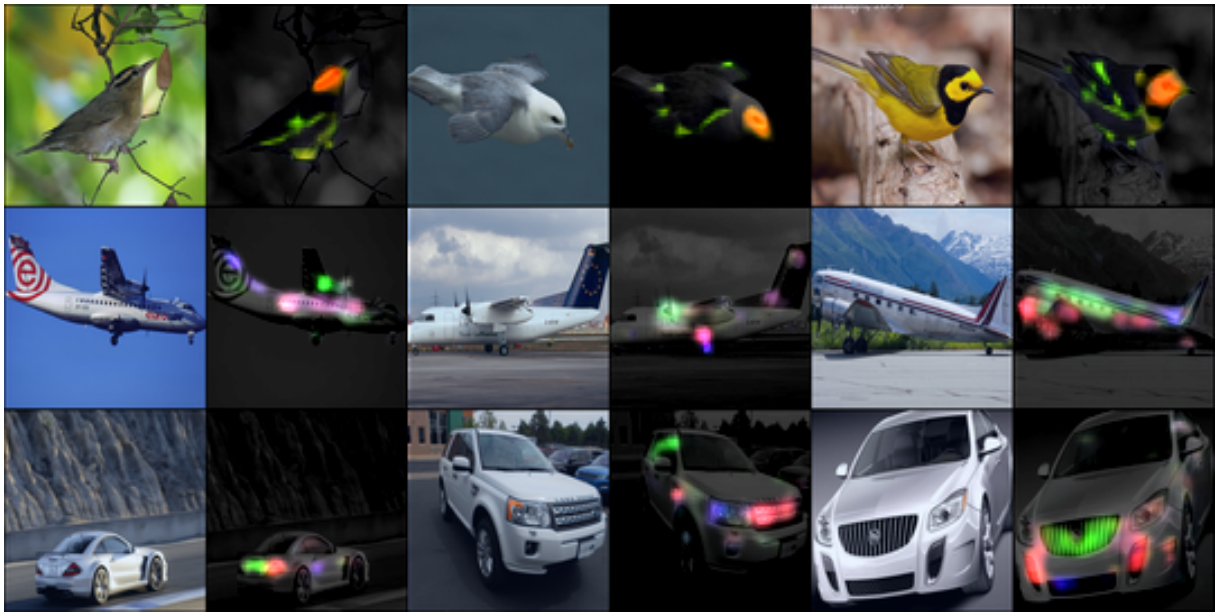


FIGURE 3.9 – Cartes d'attention produites par BR-NPA.

est 3, ce qui montre que l'utilisation de $N = 3$ n'est pas au désavantage de B-CNN.

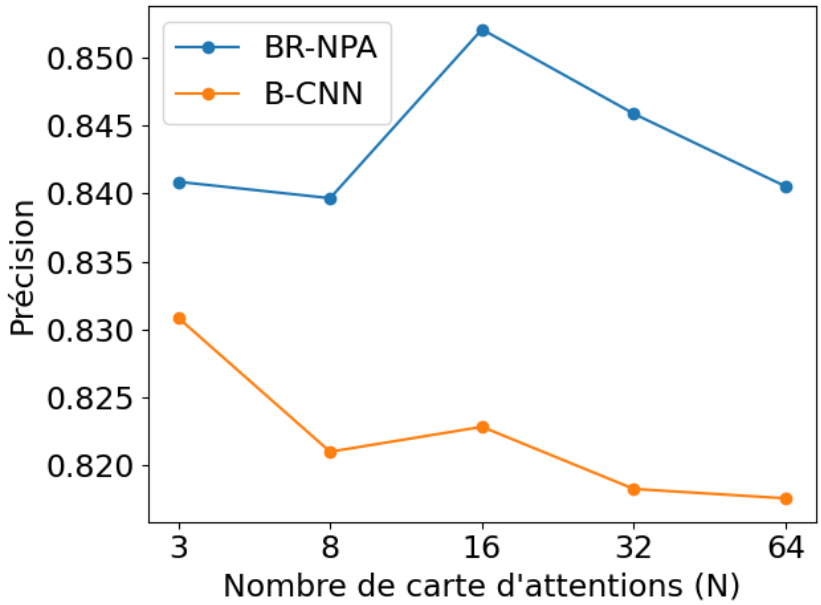


FIGURE 3.10 – Impact du nombre de cartes d'attention N sur la précision du test.

3.2.8 Stabilité de l'entraînement

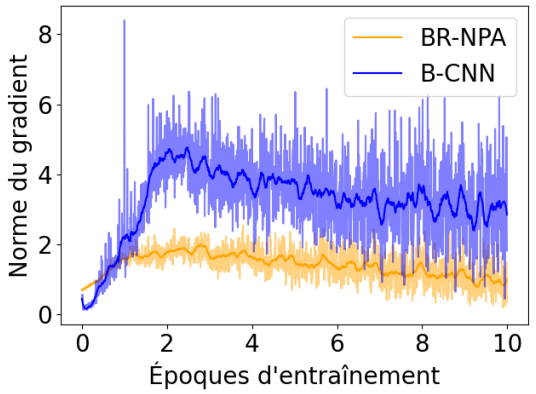
Pour évaluer l'impact de la couche d'attention non paramétrique sur les propriétés du gradient, nous proposons de visualiser la norme euclidienne du gradient des 3 dernières couches convolutives de ResNet-50 avant la couche d'attention. La figure 3.11 montre que pendant l'entraînement la norme des gradients de BR-NPA est plus faible que celle des gradients de B-CNN, ce qui permet probablement une optimisation plus efficace. Une hypothèse qui pourrait expliquer l'instabilité de l'entraînement de B-CNN est que les paramètres des couches de convolutions utilisées pour générer les cartes d'attention introduirait de la redondance dans l'espace des fonctions exprimables par le réseau. Au lieu d'avoir des minimums locaux isolés il y aurait donc des continuums de solutions où tous les points ont la même valeur de fonction de coût, ce qui rendrait l'optimisation chaotique. Par ailleurs, cette instabilité est peut-être une explication des performances supérieure de BR-NPA par rapport à B-CNN.

3.2.9 Interprétabilité des cartes de saillance produites

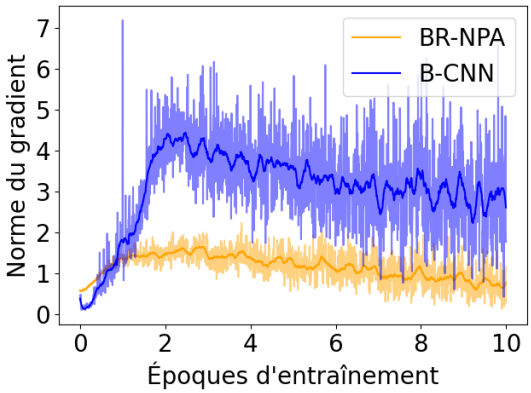
Pour évaluer l'interprétabilité des cartes d'attention générées par BR-NPA, nous utilisons les métriques introduites dans le chapitre précédent, à savoir DAUC, IAUC, DC, IC et la parcimonie.

Les valeurs moyennes des métriques sur 100 images sont indiquées par le tableau 3.7. Notez que toute les méthodes varGrad, smoothGrad et GuidedBP sont exclues de ce tableau car, générant des cartes à la résolution des images d'entrée, le nombre d'inférence et le temps de calcul nécessaire pour calculer les métriques serait trop important. On observe que le BR-NPA donne de meilleures performances que toutes les autres méthodes du point de vue des métriques DAUC, IAUC et de la parcimonie, et la deuxième meilleure performance avec DC. Comme évoqué dans le chapitre précédent, les résultats de la métrique IC sont difficiles à interpréter car toutes les valeurs sont très proches de zéro.

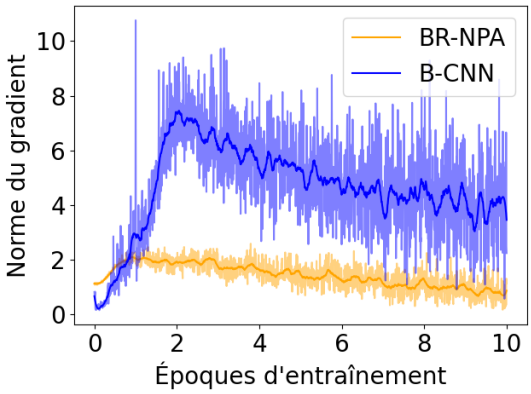
Pour vérifier si les gains de performance sont statistiquement significatifs, nous avons calculé le t-test de Welch entre les performances de chaque paire de méthodes dans la figure 3.13. BR-NPA donne des performances significativement meilleures que la plupart des autres méthodes. La première exception est RISE en termes de DAUC, ce qui peut s'expliquer par l'écart-type important qu'elle présente sur cette métrique, comme le montre la figure 3.12. La deuxième exception est ProtoTree et IBP dans IAUC, ce qui peut aussi s'expliquer par la grande variance affichée par tous les modèles sur cette métrique et le fait que ProtoTree et IBP obtiennent des performances relativement bonnes par rapport à BR-NPA. Cependant, RISE produit presque la pire performance sur la métrique IAUC et ProtoTree et IBP obtiennent également de mauvaises



(a) Couche 50



(b) Couche 49



(c) Couche 48

FIGURE 3.11 – Norme des gradients des trois dernières couches de convolution avec BR-NPA et B-CNN. Le trait foncé indique la valeur moyenne courante et le trait clair indique le signal brut.

Modèle	Visualisation	Précision	DAUC↓	IAUC↑	DC↑	IC↑	Parcimonie↑
ResNet50	Ablation CAM	84.2	0.0215	0.26	0.36	−0.04	8.54
	Grad-CAM		0.0286	0.16	0.35	−0.12	5.28
	Grad-CAM++		0.0161	0.21	0.35	−0.07	6.73
	RISE		0.0279	0.18	**0.57**	−0.11	6.63
	Score-CAM		0.0207	0.27	0.32	−0.05	5.96
	AM		0.0362	0.22	0.31	−0.09	4.04
B-CNN	-	84.8	0.0208	0.3	0.27	−0.02	12.74
BR-NPA		**85.5**	**0.0155**	**0.49**	0.41	−0.02	**16.02**
IBP		81.9	0.0811	0.48	0.23	−0.04	6.56
ProtoPNet		84.8	0.2964	0.37	0.1	−0.06	2.18
ProtoTree		82.1	0.2122	0.43	0.17	**0.04**	13.75

TABLE 3.7 – Évaluation de l'interprétabilité sur le jeu de données CUB-200-2011.

performances sur la métrique DAUC.

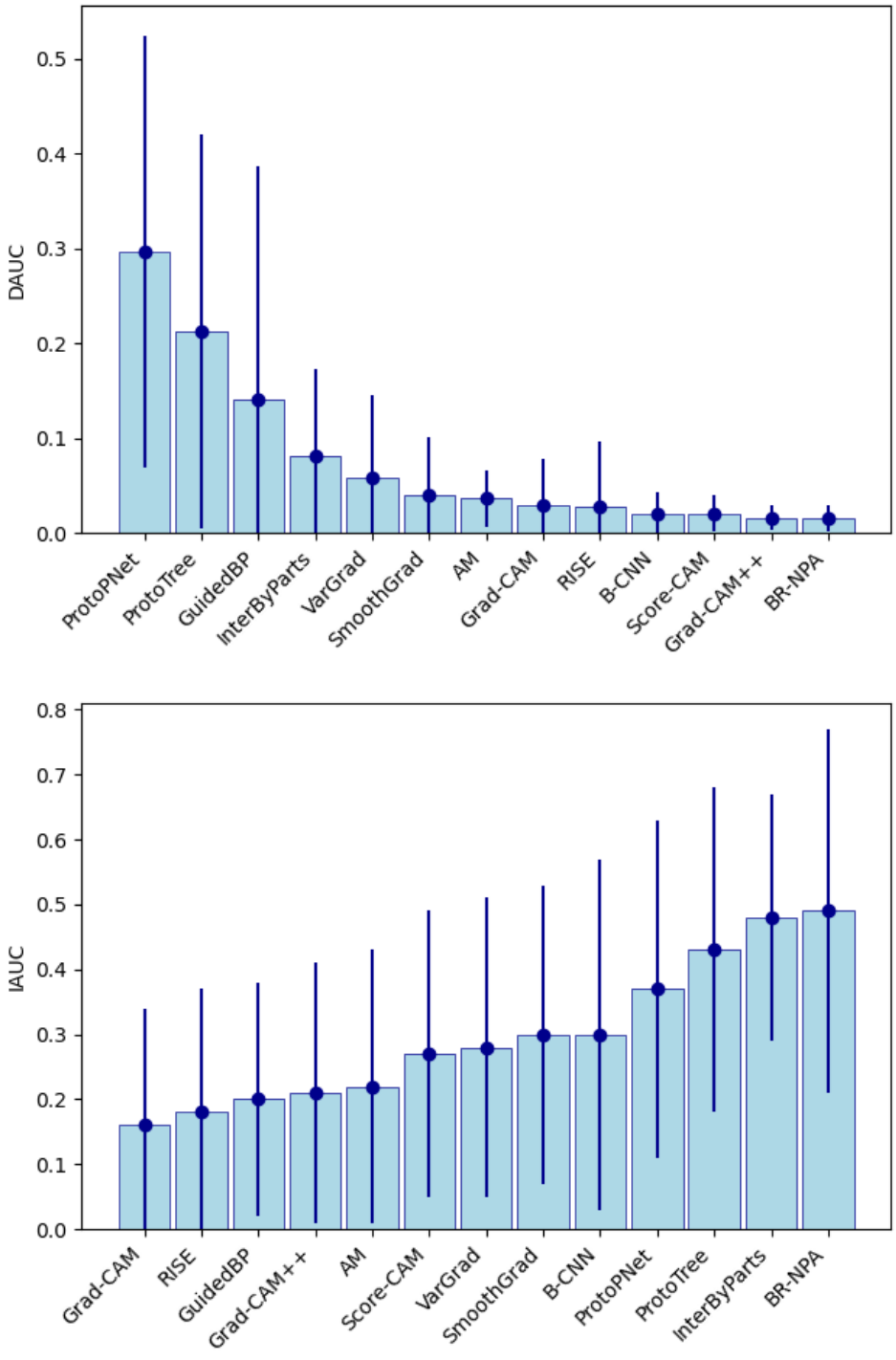
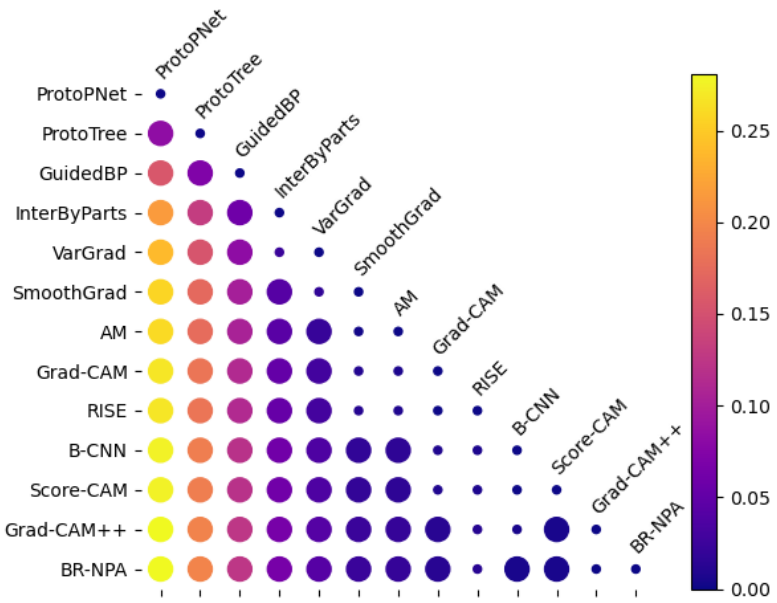
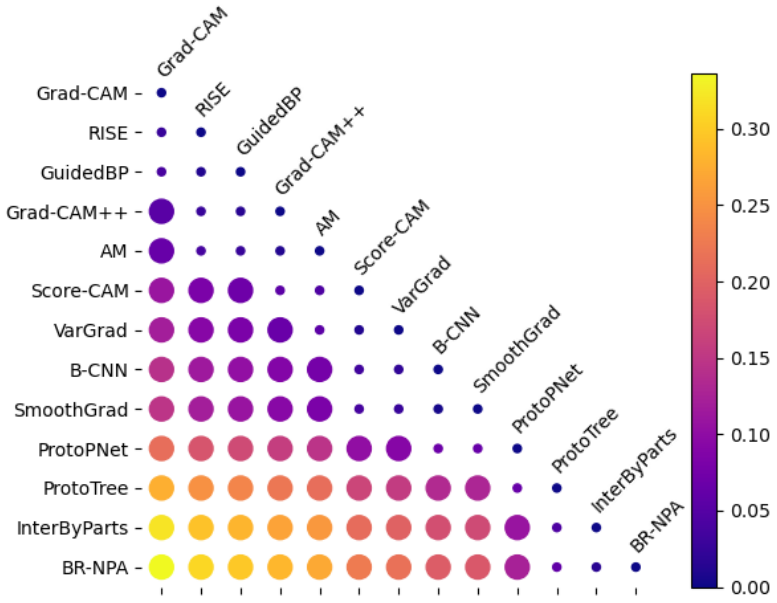


FIGURE 3.12 – Valeurs moyennes des métriques obtenues. Les modèles sont classés du pire au meilleur, de gauche à droite



(a) DAUC



(b) IAUC

FIGURE 3.13 – Comparaison du gain de performance d'une méthode à l'autre. La couleur indique la différence de performance et un grand disque indique que l'écart est statistiquement significatif (avec un critère $p_{val} < 0.05$). Les modèles sont classés du pire au meilleur, de haut en bas. À la ligne i colonne j est indiqué le gain de la méthode i par rapport à j.

3.3 Conclusion

Nous avons présenté le modèle BR-NPA, un modèle de classification proposant des cartes d'attention haute résolution à l'aide d'une couche d'attention non paramétrique. Premièrement, nous augmentons la résolution de la carte d'attention sans augmenter le taux d'erreur en classification en distillant un modèle basse-résolution dans le modèle entraîné. Deuxièmement, un algorithme non-paramétrique produit plusieurs vecteurs de caractéristiques en raffinant des vecteurs de caractéristiques saillants et dissimilaires entre eux. Ce mécanisme permet d'obtenir plusieurs cartes d'attention ordonnées par importance. La comparaison de BR-NPA avec d'autres modèle d'attention de la littérature montre que (1) BR-NPA permet d'obtenir des meilleures performances que plusieurs modèles de la littérature, notamment B-CNN, (2) peut être intégré dans des architectures pré-entraînés sans perte significative de précision de classification et (3) génère des cartes d'attention détaillées qui se focalisent sur des parties spécifiques de l'objet d'intérêt. Ces travaux ont fait l'objet d'une publication dans une revue avec comité de lecture [50].

À retenir

— Le mécanisme d'attention de BR-NPA est non-paramétrique et est basé sur l'activité et la similarité des vecteurs de caractéristiques.
— BR-NPA produit des cartes d'attention détaillées qui se concentrent sur des parties spécifiques de l'objet.
— Ce modèle permet d'atteindre des performance similaires ou supérieures aux autres modèles interprétables.

CONCLUSION DE LA PREMIÈRE PARTIE

Dans le chapitre 1, nous avons vu qu'il existe deux types d'approches pour générer des cartes de saillances. Les méthodes post-hoc sont des algorithmes qui analysent la prédiction faite par le modèle et se divisent en trois catégories : les méthodes par analyse des cartes de caractéristiques, les méthodes par gradient de l'image d'entrée et les méthodes par perturbations. Au contraire, les modèles d'attention intègrent un module d'attention au sein du modèle, le forçant ainsi à se concentrer sur une partie de l'image et à générer une carte de saillance pendant l'inférence. Nous avons distingué l'attention convolutionelle qui utilise des couches de convolutions, l'attention prototypique qui compare les vecteurs de caractéristiques à des prototypes et l'attention non paramétrique, qui n'utilise pas de paramètre entraînable dédié. Enfin, nous avons évoqué les solutions proposées dans la littérature pour évaluer l'interprétabilité des cartes de saillance et avons notamment détaillé les métriques de fiabilité. Ces métriques consistent à perturber l'image d'entrée d'un modèle en masquant ou en révélant les zones indiquées comme importantes par la carte de saillance. Les scores obtenus avec l'image perturbée sont mis en relation avec les scores de la carte de saillance pour estimer la fiabilité de la carte.

Dans le chapitre 2, nous avons analysé les métriques de fiabilité DAUC et IAUC et montré qu'elles ont en particulier deux limites. La première est qu'elles évaluent les approches en se basant sur un comportement du modèle qui n'est pas fiable, car les images perturbées ne font pas partie de la distribution d'entraînement, ce qui nuit probablement à la qualité de l'estimation. La seconde limite est que ces métriques ne tiennent pas compte des scores de saillance eux-mêmes et ne considèrent que l'ordre des scores de saillance. Cela conduit les métriques à attribuer le même score de fiabilité à des cartes de saillance aux apparences très différentes. Pour pallier ce problème, nous avons proposé trois nouvelles métriques. La première est appelée *parcimonie* et mesure à quel point une carte de saillance est concentrée sur un (ou des) points particulier(s) de l'image. Les deux autres, appelées DC et IC, mesurent la corrélation linéaire entre les scores de saillance et l'impact de chaque zone sur le score de la classe.

Dans le chapitre 3, nous avons proposé un modèle d'attention non paramétrique conçu pour l'interprétabilité. Jusqu'à présent, les seules approches pensées pour être interprétables à notre connaissance sont des modèles prototypiques. De plus, il n'y a eu jusqu'à maintenant que peu de travaux à propos de mécanismes d'attention non paramétriques. Ce modèle est appelée

BR-NPA et calcule plusieurs cartes d'attention par image en utilisant l'activité des vecteurs de caractéristiques pour repérer les différentes parties de l'objet et la similarité cosinus avec les voisins pour raffiner les vecteurs de caractéristiques obtenus. La colonne vertébrale utilisée est également modifiée pour générer des cartes d'attention avec une résolution quatre fois plus grande qu'avec la résolution par défaut. En utilisant diverses évaluations incluant notamment des métriques de fiabilité, nous avons montré que BR-NPA génère des cartes d'attention fiables tout en proposant de meilleures performances de classification que les autres modèles interprétables et méthodes post-hoc étudiées. De plus l'observation des cartes de BR-NPA montrent que ce modèle focalise précisément son attention sur des parties de l'image, permettant de voir qu'il utilise des indices visuels pertinents pour la tâche à accomplir.

Dans la partie suivante nous introduirons les données issues de l'imagerie time-lapse et la tâche de prédiction des paramètres MC. Nous constaterons en chapitre 4 l'absence de travaux en explicabilité sur les vidéos time-lapse malgré un besoin important de la communauté dans ce domaine. Nous proposerons donc une base de référence publique pour cette tâche en chapitre 5 et en chapitre 6 nous y appliquerons plusieurs modèles interprétables (dont BR-NPA) et méthodes d'explication dont nous évaluerons l'interprétabilité avec des métriques de fiabilité.

DEUXIÈME PARTIE

Applications aux vidéos time-lapse d'embryons

CHAPITRE 4

ÉTAT DE L'ART DES APPLICATIONS DE L'APPRENTISSAGE PROFOND AUX IMAGES ISSUES DE SYSTÈMES TIME-LAPSE

Dans ce chapitre, nous dressons un état de l'art des applications de l'apprentissage profond (AP) aux données issues de systèmes d'imagerie Time-lapse ("Time-Lapse Imagery", TLI) utilisés pour la Fécondation In Vitro (FIV). Nous nous intéressons plus particulièrement à la prédiction de la qualité embryonnaire et à la détection des paramètres morpho-cinétiques (MC). Nous commençons par décrire ce que sont les vidéos time-lapse d'embryons et leurs annotations puis nous détaillons les travaux récents à propos de la prédiction de la qualité embryonnaire et de l'annotation automatique des paramètres MC.

4.1 Les vidéos time-lapse d'embryons

Nous avons mentionné dans l'introduction que les incubateurs à TLI permettent un suivi continu du développement de l'embryon, en prenant des photos de chaque embryon à intervalles réguliers tout au long de son développement, pour finalement compiler une vidéo donnant un aperçu dynamique du développement embryonnaire in vitro. Les vidéos produites par ces incubateurs montrent donc chacune le développement en accéléré d'un embryon. Une vidéo contient en moyenne 500 images en nuances de gris avec une résolution de 500 par 500 pixels. Les vidéos sont enregistrées selon plusieurs réglages du plan focal du microscope ce qui permet une vision quadridimensionnelle de l'embryon : les trois dimensions spatiales et la dimension temporelle, comme illustré en figure 4.1. Un exemple de vidéo time-lapse est disponible ici.

Ces vidéos peuvent être accompagnées par plusieurs types d'annotations. Nous nous concentrons dans ce manuscrit sur les paramètres morpho-cinétiques (MC) mais nous évoquons aussi les notes de la masse cellulaire interne ("Inner Cellular Mass", ICM) et du Trophectoderme (TE) ainsi que l'issue de la FIV.

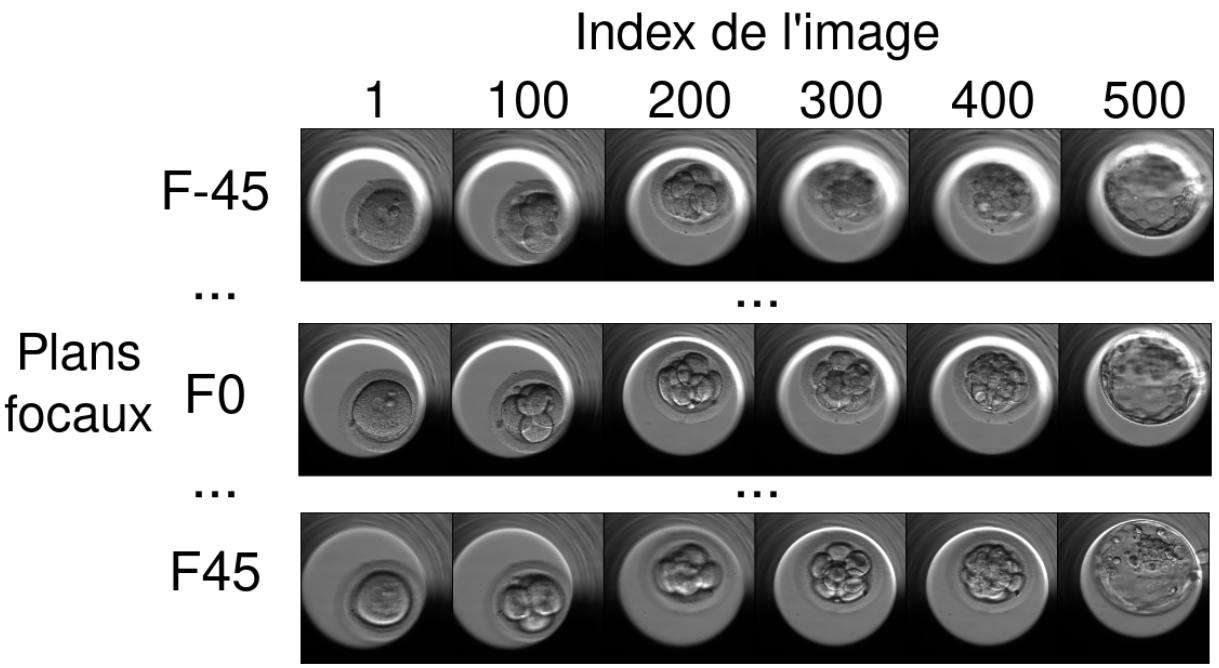


FIGURE 4.1 – Illustration d'une vidéo produite par un incubateur TLI (Embryoscope©, Vitrolife©, Sweden).

4.1.1 Paramètres MC

Les paramètres MC consistent en la mesure du temps séparant 16 événements cellulaires successifs. Dans la suite de ce manuscrit, nous utiliserons le mot 'instant' pour indiquer cette mesure temporelle. Ces événements sont notés tPB2, tPNa, tPNf, t2, t3, t4, t5, t6, t7, t8, t9+, tM, tSB, tB, tEB, et enfin tHB. Nous utilisons la définition des événements proposée par Ciray et al. [25] : apparition du deuxième globule polaire (tPB2), apparition et disparition des pronucléi (tPNa et tPNf), division des blastomères à partir du stade de deux cellules au stade de neuf cellules (et plus) (t2,t3,t4,t5,t6,t8 et t9+), compaction (tM), formation du blastocyste (tSB, tB), expansion et éclosion (tEB et tHB). Notez que tous les embryons ne parviennent pas à développer jusqu'au stade tHB et peuvent se nécroser (c-à-d. mourir) à tout moment du développement. En particulier, on appelle "blastulation" (Blast.) l'évènement tB durant lequel l'embryon devient un blastocyste. Ces différents stades de développement sont illustrés en figure 4.2. Notez que, selon leur position dans le puits dans lequel ils se trouvent, les embryons peuvent parfois être partiellement occultés, ce qui est assez courant dans les vidéos time-lapse. Cependant, même lorsqu'une partie de l'embryon est cachée, la partie visible est suffisante pour identifier la phase de développement.

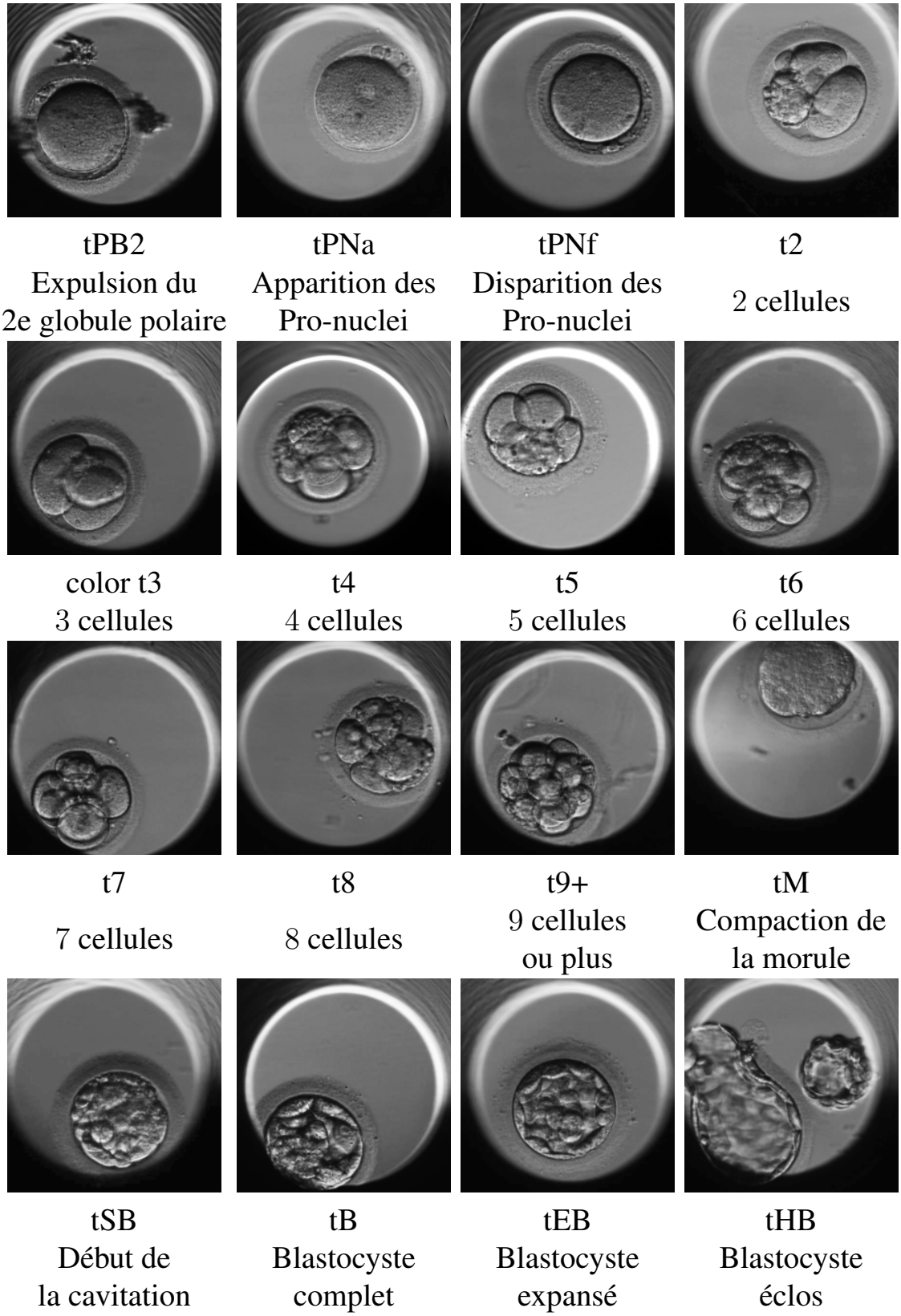


FIGURE 4.2 – Illustrations des 16 phases de développements morpho-cinétiques utilisées.

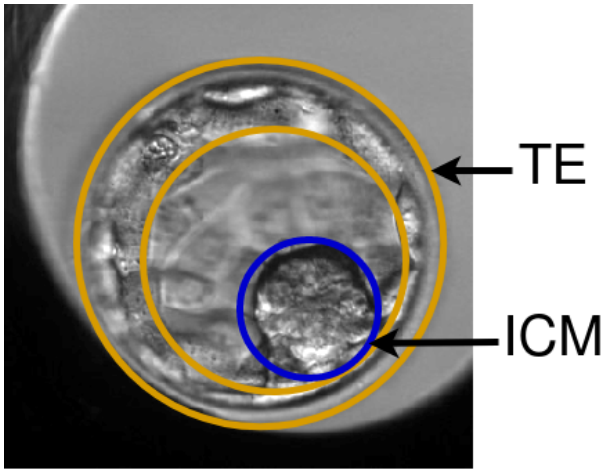


FIGURE 4.3 – Le TE et l'ICM sont deux parties de l'embryon qui apparaissent au stade de blastocyste.

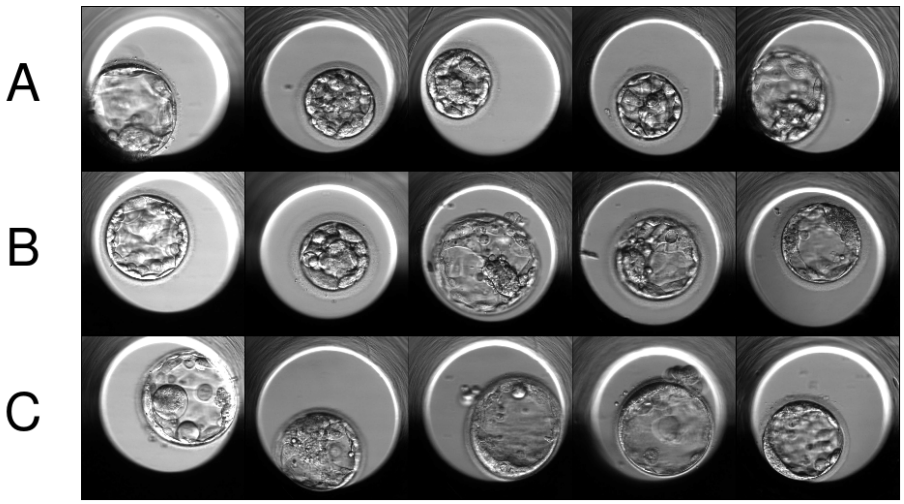
4.1.2 Les notes du TE et de l'ICM

L'une des tâches quotidiennes des embryologistes est de classer les embryons au stade de blastocyste à l'aide d'un microscope en suivant le système de classement de Veeck et Zaninovic [149]. Ce système propose plusieurs critères morphologiques pour évaluer la qualité du trophectoderme (TE) et de la masse cellulaire interne (ICM), qui sont deux parties de l'embryon au stade de blastocyste, comme illustré en figure 4.3.

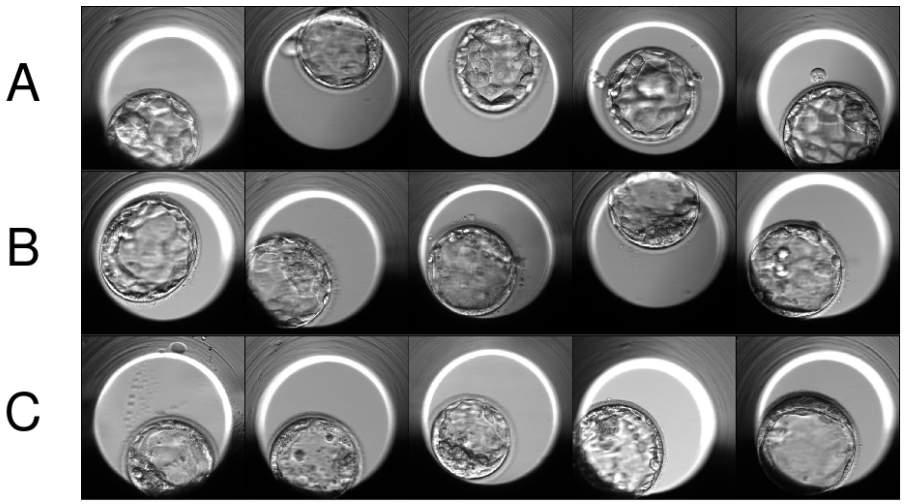
Chaque partie est notée par une lettre A, B ou C, qui représente respectivement une bonne, une moyenne et une mauvaise qualité. L'ICM et le TE sont annotés séparément et peuvent recevoir des notes différentes, c-à-d. qu'un embryon peut avoir une ICM et un TE de qualité différente. Ce système aide les biologistes à sélectionner l'embryon le plus approprié pour le transfert, car il a été démontré que les critères utilisés sont corrélés au taux de réussite de la FIV [81]. La figure 4.4 montrent des exemples d'embryons ayant reçu les notes A, B et C.

4.1.3 L'issue de la FIV

Il existe plusieurs variables pouvant indiquer l'issue de la FIV. Celles-ci se distinguent par le moment de la grossesse durant lequel la variable est connue. On distingue l'implantation (Impl.) de l'embryon, qui indique si l'embryon s'est implanté ou non dans l'utérus, puis le test de grossesse (Gross.), ensuite la présence ou non d'un battement de coeur (BC) du foetus et enfin la naissance vivante (NV) ou non. Un propriété importante est que la négativité d'une de ces variables entraîne la négativité des suivantes. Par exemple si l'implantation ne réussit pas, le test



(a) ICM



(b) TE

FIGURE 4.4 – Illustration des trois notes pouvant être attribuées à l'ICM et au TE d'un embryon. Ces notes sont attribuées en fonction de critères morphologiques définis par Veeck et al. [149].

de grossesse sera par la suite négatif et aucun battement de coeur ne sera détecté. De même, si le test de grossesse est négatif, il est très probable qu'on ne détecte pas de battement de coeur ensuite.

4.2 État de l'art

Dans cet état de l'art nous évoquons d'abord les principaux outils disponibles sur le marché pour le traitement des images issues de systèmes TLI. Ensuite, nous dressons une revue des travaux de recherche dans le domaine de l'exploitation des images de TLI par l'AP. Plus précisément, nous évoquons dans un premier temps les solutions proposées par les auteurs pour les problèmes de prédiction de la qualité embryonnaire. Nous nous concentrons ensuite sur les solutions pour automatiser l'annotation des paramètres MC car c'est sur cette tâche que seront entraînés les modèles dans la suite de ce manuscrit.

4.2.1 Outils existants sur le marché

De nombreuses entreprises proposent actuellement des outils logiciels basés sur de l'apprentissage machine pour répondre à des problématiques de FIV. Étant donné que ces modèles sont implémentés dans des produits vendus par ces entreprises, ces dernières ne communiquent pas à propos de l'entraînement ou de l'architecture des modèles. Cependant, les entreprises publient des évaluations de leur modèles, en comparant leur performance avec celle des biologistes. Nous proposons donc dans le tableau 4.1 une revue succincte de ces outils, en se basant sur ces publications. Les résultats obtenus par ces outils montrent qu'il est difficile de prédire si un embryon va mener à un test de grossesse positif ou non à partir de la vidéo time-lapse. Par exemple, avec CHLOE© [166], IdaScore© et KIDScore© [145], on obtient une aire sous la courbe ("Area Under Curve", AuC) de la fonction d'efficacité du récepteur ("Receiver Operating Curve", ROC) de respectivement 0.61, de 0.66 à 0.76 et de 0.66 à 0.75. De façon similaire, le modèle de LifeWhisperer© [150] obtient une sensitivité et une spécificité respectivement de 0.70 et 0.60. Il est plus facile de prédire si l'embryon va devenir un blastocyste (CHLOE atteint une AuC de 0.96) et il est même possible d'opérer une sélection avant la fécondation. L'outil Magenta permet en effet de sélectionner les ovocytes les plus prometteurs et on observe une différence significative de blastulation entre les ovocytes les mieux notés par Magenta [40] (46.1%) et les ovocytes les moins bien notés (26.6%). Il est aussi possible d'entraîner un modèle à annoter les notes de qualité données par les biologistes à l'ICM et au TE des blastocystes. Ces

notes sont utilisées pour identifier les embryons avec le meilleur potentiel et ceux qui ont les plus hautes notes ont donc le plus de chance d'être transféré. Par exemple, l'outil EMA© [111] est entraîné à prédire ces notes et permet de calculer la probabilité qu'un embryon soit transféré. En effet, EMA donne les plus hautes notes à 100% des embryons transférés.

La difficulté de ces outils à prédire le résultat du test de grossesse vient du fait que l'issue de la FIV ne dépend pas que de l'embryon mais aussi de facteurs utérins, qui sont inaccessibles au modèle (et à un biologiste) en observant seulement la vidéo time-lapse de l'embryon.

	Prédiction(s)	Données de test	
		Nb. de Cliniques	Nb. d'embryons
CHLOE© [166]	paramètres MC blastulation Gross.	1	6748 (877 transf.)
IdaScore©, KIDScore© [145]	Gross.	1	3014
LifeWhisperer AI [150]	Gross.	11	8886
EMA© [111]	Notes de qualité	Non indiqué	Non indiqué
Oocyte Magenta© [40]	Blast.	1	392

TABLE 4.1 – Les principaux outils d'AP disponibles sur le marché pour traiter les images issues de la FIV. La prédiction de la blastulation (Blast.) consiste à prédire si l'embryon va atteindre la phase tB de son développement et ainsi devenir un blastocyste. Les notes de qualité sont déterminées selon des critères morphologiques proposés par Veeck et Gardner [149]. Notez que les auteurs de l'évaluation de CHLOE ne reportent pas l'évaluation de la prédiction des paramètres MC.

4.2.2 Prédiction de la qualité embryonnaire

Dans cette section nous passons en revue les travaux récents sur la prédiction de la qualité embryonnaire. Il existe plusieurs variables indiquant la qualité embryonnaire qui sont pertinentes à prédire. Parmi elles, on trouvent les variables indiquant l'issue de la FIV (Impl., Gross., BC, et NV) auquel on ajoute la blastulation (Blast.) et le nombre de pro-nucléi (PN) qui doit être égal à deux pour permettre un bon développement. Le tableau 4.2 indique les principales caractéristiques des travaux sur la prédiction de la qualité embryonnaire de 2020 à aujourd'hui.

D'abord, on remarque que presque tous les travaux utilisent un CNN pour traiter les images et un LSTM [61] ou un CNN 3D [55] pour exploiter le contexte temporel de la vidéo, ce qui n'est pas surprenant étant donné que ces architectures dominent actuellement (avec les transformers) les bases de référence les plus importantes en apprentissage machine. La seule approche pouvant

Auteurs	Tâche ou Prédiction	Nb. d'embryons	Modèle	AuC
Fukunaga et al. [44]	Comptage des PN	900	CNN	-
Xie et al. [163]	Blast.	3k	CNN+attention	0.77
Liao et al. [91]		10k	CNN+LSTM	0.82
Silver et al. [132]	Impl.	272	CNN+LSTM	-
Abbasi et al. [1]		130	CNN	-
Chavez-Badiola et al. [20]	Gross.	100	SVM/RF	0.75 à 0.77
Berntsen et al. [13]	BC	14k	CNN 3D	0.67
Tran et al. [143]		9k	CNN	-
Campbell et al. [17]	NV	63k	CNN	0.69
Huang et al. [66]		101	Non indiqué	-
Sawada et al. [125]		470	CNN +attention	-
Huang et al. [65]		6k	CNN	-

TABLE 4.2 – Auteurs, tâche, taille du jeux de données, modèle et performance obtenues des travaux sur la prédiction de la qualité embryonnaire de 2020 à aujourd'hui. Les auteurs de Huang et al. [66] ne reportent pas le type de modèle utilisé. Les auteurs qui ne reportent pas l'AuC utilisent d'autres métriques et les performances qu'ils obtiennent sont discutées en section 4.2.2.

être qualifiée d'interprétable dont nous ayons connaissance est le travail de Sawada et al. [124] qui ont appliqué un modèle d'attention au problème de la prédiction de la NV. L'architecture employée est un modèle ABN, évoqué dans le chapitre 1. Pour rappel, le module d'attention convolutionel du modèle ABN est constitué de plusieurs convolutions et est appliqué de façon résiduelle aux cartes de traits. Lors d'une étude qualitative des cartes d'attention obtenues, les auteurs n'observent pas de lien entre les zones d'attention obtenues et l'issue de la FIV. Ils notent cependant que pour de nombreuses images l'attention est focalisée autour de la zone pellucide [1]. On peut aussi citer Xie et al. [164] qui ont proposé d'utiliser le module d'attention spatial non-local de Wang et al. [158] pour prédire la blastulation. Étant donné qu'il s'agit d'un module non-local, similaire à ceux utilisés dans un transformer, il est difficile à interpréter car il génère un grand nombre de cartes d'attention (une par vecteur de caractéristiques) et nous ne le détaillons donc pas ici.

Notez qu'il n'existe pas de consensus quant aux métriques à utiliser pour reporter la performance des modèles. Ainsi, la métrique la plus répandue est l'AuC de la courbe ROC mais celle-ci n'est pas utilisée dans la majorité des travaux. Parmi les auteurs qui reportent l'AuC, on

1. Un manteau extra-cellulaire relativement épais qui entoure l'embryon.

note la faible valeur des performances obtenus pour la prédiction de la NV et du BC (0.69 et 0.67). Comme évoqué précédemment, cela s'explique par le fait que ces variables dépendent de l'embryon mais aussi largement de facteurs utérins qui ne sont pas possibles à mesurer en ayant seulement accès à la vidéo. D'autres auteurs ont donc proposé de donner des informations sur ce type de facteur au modèle en plus de la vidéo ce qui permet d'obtenir un AuC de 0.75 à 0.77 pour la prédiction du test de grossesse [20]. Par exemple, Abbasi et al. [1] reportent un taux de rappel de 0.80 et une précision de 0.76 pour les embryons qui parviennent à s'implanter et Silver et al. [132] reportent une valeur prédictive positive (VPP) de 0.93 et une valeur prédictive négative (VPN) de 0.58 pour la même classe. Enfin, Fukunaga et al. [44] entraînent un modèle à compter le nombre de PN et reportent une sensitivité de 0.99, 0.82, et 0.99 lorsqu'il y a respectivement 1, 2 et 3 PN. Parmi les autres méthodes d'évaluation, Sawada et al. [125] calculent la distribution des scores de qualité donnés par le modèle et montrent que le score médian des embryons ayant abouti à une NV est significativement plus élevé que celui des embryons ayant abouti à une naissance non vivante.

Étant donné que la majorité des embryons cultivés ne sont pas transféré, les variables Impl., Gross., BC et NV ne peuvent être connues que pour un sous-ensemble des embryons cultivés. Ce sous-ensemble n'est pas représentatif de la population générale des embryons, car il correspond aux embryons transférés, c-à-d. les embryons avec la plus haute qualité selon les biologistes. Il y a donc un biais de sélection sur cet ensemble et un modèle entraîné avec pourrait mal généraliser sur la population entière des embryons. Certains auteurs ont donc proposé de faire l'hypothèse que les embryons non-transférés qui sont détruits par les biologistes n'auraient pas mené à une grossesse s'ils avaient été transférés. Cette solution permet de résoudre le problème du biais de sélection en incluant ces embryons dans le jeu de données. Cela a pour conséquence d'accroître notablement les valeurs d'AuC obtenues avec cette hypothèse : Berntsen et al. [13], Tran et al. [143] et Huang et al. [65] obtiennent respectivement 0.95, 0.93 et 0.97. On peut néanmoins questionner la validité de cette hypothèse dans la mesure où les critères de sélection des embryons utilisés par les biologistes pourraient être améliorés et sont sujets à de la variabilité inter et intra-opérateur [35]. Il est donc possible qu'une partie des embryons détruits auraient menés à des grossesses, voir à des naissances en vie.

4.2.3 Reconnaissance des paramètres MC

Nous dressons ici un état de l'art des solutions proposées pour l'annotation automatique des paramètres MC. Nous évoquons d'abord les caractéristiques des jeux de données et les phases de développement utilisées par les auteurs. Ensuite, nous décrivons les méthodes utilisées pour

améliorer la performance des modèles sur cette tâche. Enfin, nous évoquons une contribution effectuée dans le cadre de ce projet sur les mêmes données avant le début de cette thèse.

Caractéristiques des jeux de données

Nous avons répertorié dans le tableau 4.3 les principales caractéristiques des jeux de données utilisés pour l'extraction de paramètres MC. On note d'abord que presque la moitié des travaux n'utilisent pas de vidéos mais seulement des images isolées. De plus, on observe que la plupart des travaux précédents ont proposé d'entraîner un modèle à ne distinguer qu'un nombre réduit de stades de développement. Par exemple, la plupart des auteurs proposent de n'utiliser que les premiers stades de développement [80, 118, 131, 159, 87, 97, 109]. On peut aussi citer Kanakasabapathy et al. [78], qui ont proposé un modèle entraîné à détecter l'instant auquel un embryon devient un blastocyste. Ce choix de ne considérer qu'un nombre réduit de phases de développement est motivé par la grande variabilité intra- et inter-opérateur qui existe au sein des annotations des paramètres MC [78].

Par exemple, les phases tardives (tM, tSB, tB et tEB) ont un écart type intra-opérateur 3 à 5 fois plus élevé que les phases de divisions cellulaires précoces [102]. Cette variabilité peut rendre l'apprentissage du modèle difficile et diminue la fiabilité de l'évaluation. Fusionner ou ignorer des stades de développement permet donc de pallier ce problème de variabilité. On peut néanmoins citer les travaux récents de Campbell et al. [17], Leahy et al [88] et de Feyeux et al. [39] qui utilisent de plus grand ensembles de phases de développement (respectivement 12, 11 et 16 phases). Nous discutons plus particulièrement de l'approche de Feyeux et al. en section 4.2.3 car, contrairement aux autres modèles qui sont des DNN, il s'agit d'un algorithme *ad-hoc*.

Modèles utilisés

Exploitation du contexte temporel. Plusieurs travaux ont proposé d'exploiter le format vidéo des données en exploitant les images voisines de l'image d'intérêt afin d'améliorer la performance des modèles. La plupart des études ont proposé des méthodes de fusion tardive en calculant les mêmes caractéristiques sur plusieurs images successives, puis en agrégeant les informations pour produire une prédiction. Par exemple, il a été proposé de concaténer les vecteurs de caractéristiques des images calculées par le même réseau neuronal convolutif (CNN) [109], ou de les agréger en utilisant le max-pooling [97]. Enfin, Lau et al. utilisent un réseau récurrent LSTM (Long-Short Term Memory) [61] pour prendre en compte les caractéristiques extraites de

Auteur	Année	Type	Taille du jeu d'entraînement		Stades de développement	Précision
			Nb. de vidéos	Nb. d'images	à distinguer (Nb.)	obtenue (%)
Khan et al. [80]	2016	Vidéos	256	150k	tPNf à t5 (5)	87
Moradi Rad et al. [118]	2018	Images	-	224	tPNf à t5 (5)	82.4
Silva-Rodríguez et al. [131]	2019	Vidéos	263	100k	tPNf à t5 (5)	80.9
H Ng et al. [109]	2018	Images	-	600k	tPB2 à t4+ (6)	84.6
Liu et al. [97],	2019	Vidéos	170	60k	tPB2 à t4+ (6)	83.8
Lau et al. [87]	2019	Vidéos	1303	145k	tPB2 à t4+ (6)	83.65
Kanakasabapathy et al. [78]	2019	Images	-	8k	pré/post tB (2)	96
Campbell et al. [17]	2022	Vidéos	63k	4M	tPNa à tB (12)	Non indiqué
Leahy et al. [88]	2020	Images	341	170k	tPNf à tB (11)	81.2
Feyeux et al. [39]	2020	Vidéos	746	373k	tPB2 à tHB (16)	58.0

TABLE 4.3 – Caractéristiques des jeux de données et performances obtenues par les travaux précédents.

chaque image [87]. On note que la fusion d'information précoce a également été proposée en concaténant les images dans la dimension des canaux [109]. Pour exploiter le contexte temporel, nous utiliserons également des LSTM mais nous proposons également d'utiliser ResNet-3D, un type de CNN spécifiquement conçu pour le traitement vidéo.

Plausibilité biologique des prédictions. Les différentes phases du développement sont strictement ordonnées : lors de son développement, un embryon ne peut pas revenir à une phase qui a précédée celle à laquelle il se trouve. Dû à la formulation du problème de classification d'images, les modèles de type DNN n'ont pas cette contrainte et prédisent donc des séquences d'évènements qui sont quasi-systématiquement impossibles en réalité. Dans la littérature, la solution actuellement utilisée est inspirée de la programmation dynamique. Plus précisément, on utilise l'algorithme de Viterbi pour croiser les scores de probabilités extraits par le DNN avec les séquences qui sont biologiquement possibles [80, 87, 97, 109, 159]. Pour chaque image, le modèle produit une distribution indiquant, pour chaque phase possible, la probabilité que l'image appartienne à cette phase. L'algorithme de Viterbi traite ces distributions et, en tenant compte du fait que les phases sont ordonnées, produit une prédiction pour chaque image telle que la séquence des phases prédites le long de la vidéo est cohérente d'un point de vue biologique et optimale du point de vue des scores de classification [151].

Modèle *ad-hoc*. Cette thèse s'inscrit dans le cadre du projet DL4IVF au sein duquel d'autres travaux ont été effectués avant cette thèse. Le travail de Feyeux et al. [39] propose une méthode *ad-hoc* basée sur des méthodes de traitement d'images traditionnelles. La méthode consiste

d'abord à localiser l'embryon avec une transformée de Hough et à identifier l'heure à laquelle a été prise chaque image en utilisant une méthode de reconnaissance de caractères sur l'heure affichée en bas de l'image. Les premiers stades de développement sont identifiés grâce à l'analyse de la variation des niveaux de gris de l'image et les derniers stades sont obtenus par seuillage itératif de l'entropie de l'image. Le tableau 4.3 montre que cette approche obtient de moins bonnes performances que les DNN, démontrant l'intérêt de ces derniers pour cette tâche.

4.2.4 Conclusion

Dans l'ensemble, les applications actuellement existantes d'AP aux images de TLI sont concentrées à entraîner des architectures pré-existantes avec des procédures standards. On peut cependant distinguer deux innovations mineures proposées sur le problème de reconnaissance des paramètres MC. La première se situe dans l'exploitation de l'ordonnancement des classes du problème. Celles-ci ne peuvent se succéder que dans un certain ordre et cette information peut être exploitée à l'aide de méthodes dynamiques tel que l'algorithme de Viterbi. Cela permet de réduire le taux d'erreur du modèle en éliminant les séquences biologiquement impossibles. La seconde innovation se trouve dans l'exploitation du contexte temporel. Le contexte temporel est souvent utilisé en apprentissage machine pour des tâches où les données sont sous format vidéo comme la reconnaissance d'action. Cependant, le problème de la prédiction des paramètres MC diffère notablement de la plupart des tâches mettant en jeu des vidéos étudiées en apprentissage machine. En effet, ces tâches requièrent du modèle une seule prédiction par vidéo alors que dans le cas de la classification des stades de développement, le modèle doit fournir une prédiction pour chaque image de la vidéo. Les propositions des auteurs pour exploiter ce contexte sont donc spécifiques à ce problème et peuvent donc être considérées comme des innovations. Dans l'ensemble, ces innovations sont mineures car elles sont largement basées sur des méthodes proposées précédemment dans le domaine de l'apprentissage machine. Les précédentes applications d'AP à cette tâche n'ont donc proposé que peu d'innovations du point de vue méthodologique.

Du point de vue des données, ces travaux souffrent aussi de certaines limites. Premièrement, toutes les études disponibles n'ont utilisé que des ensembles de données privés, ce qui rend impossible la reproduction des résultats de ces études. Deuxièmement, ces études n'utilisent souvent qu'un nombre restreint de stades de développement, à savoir les premières divisions cellulaires. Les modèles entraînés ne permettent donc d'annoter l'embryon que pendant une partie réduite de son développement, ce qui limite leur intérêt. Notez que l'outil CHLOE, développé par l'entreprise Fairtility©, que nous avons mentionné précédemment, permet d'annoter un plus grand nombre de stades de développement (à partir de tPNa jusqu'à tEB). Cependant, cet

outil est issus de recherches privées, et la méthodologie d'entraînement de ce modèle n'est pas communiqués par Fairtility©. Pour cette raison, nous n'avons pas pu inclure ce modèle dans cet état de l'art.

Nous proposons donc dans le chapitre suivant une base de référence publique de vidéos time-lapse avec un grand nombre de phases annotées. De plus, les données sont disponibles sous la forme de vidéos complètes avec tous les plans focaux. En effet, les travaux précédents ont montré l'intérêt d'exploiter le contexte temporel pour réduire le taux d'erreur du modèle. Il est donc très pertinent de proposer un jeu de données de grande taille constitué de vidéos complètes permettant de bénéficier du contexte temporel. On peut supposer que proposer tout les plans focaux disponibles pourra permettre d'exploiter le contexte spatial et d'améliorer encore les performances.

Face au nombre réduit de modèles interprétables proposés dans le domaine de la FIV, nous proposons également en chapitre 6 d'appliquer le modèle interprétable développé dans le chapitre 3 sur cette tâche et de comparer sa performance à d'autres modèles en utilisant notamment les métriques développées en chapitre 2.

À retenir

— Les données embryonnaires sont des vidéos qui montrent le développement d'un embryon en accéléré.
— Elles peuvent être accompagnées de plusieurs annotations : les durées des phases de développement de l'embryon, la qualité de son ICM/TE et enfin l'issue de la FIV s'il a été choisi pour être transféré.
— Jusqu'à présent, il n'y a pas eu de travaux spécifiquement concentrés à évaluer l'interprétabilité de modèles entraînés sur ces données.

CHAPITRE 5

APPLICATION DU JEU DE DONNÉES EMBRYONNAIRES À LA RECONNAISSANCE DE PARAMÈTRES MC

Dans ce chapitre nous présentons une base de référence publique pour entraîner des modèles à la reconnaissance de paramètres MC. Pour cela nous utilisons une partie du jeu de données décrit précédemment, choisie spécialement pour entraîner des modèles à la prédiction des paramètres MC. Les données et les annotations sont accessibles ici [49]. En effet, bien qu'il existe un consensus pour l'évaluation morphologique du développement embryonnaire, cette méthode souffre encore d'un manque de pouvoir prédictif et d'une variabilité inter- et intra-opérateur [27, 139, 11]. Automatiser cette tâche permettrait donc pour réduire cette variabilité. Aussi, il est important de noter que la question du partage des données est au centre des stratégies d'AP appliquées aux données de santé. En effet, un modèle ne peut pas être reproduit et re-évalué si le jeu de données utilisé pour entraîner le modèle n'est pas disponible au public. La principale raison de cette absence fréquente de partage des données est probablement liée à des préoccupations concernant la sécurité des données et peut-être, dans une moindre mesure, à la concurrence scientifique. La conséquence d'un développement plutôt "boîte noire" des méthodes d'AP dans le domaine de la FIV est l'absence de consensus sur l'architecture d'AP à utiliser, avec des sociétés privées qui vendent et mettent en œuvre des solutions qui n'ont pas été évaluées de manière indépendante par la communauté, ce qui soulève notamment des questions sur les problèmes potentiels de partialité et d'équité [4]. Le partage des données est donc de la plus haute importance pour mettre en œuvre correctement l'AP dans la pratique de la FIV [4]. Dans ce contexte, nous avons grandement besoin d'un ensemble de données de référence time-lapse, à l'instar de ce qui a été fait dans d'autres domaines [76, 116, 138, 75].

Comme on l'a vu dans le chapitre précédent, plusieurs équipes ont appliqué des modèles d'AP à la FIV, mais avec des limitations importantes : soit le nombre de vidéos était inférieur à 300, soit le nombre total d'images composant les vidéos était inférieur à 150k [87, 131, 80, 118]. De

plus, ces études ont utilisé un nombre limité de stades embryonnaires (i.e. de paramètres MC) à identifier. Enfin, et comme indiqué ci-dessus, ces études n'ont pas partagé leurs données, rendant leur travaux impossibles à reproduire. Un jeu de données partagé devrait être suffisamment grand pour entraîner de puissants modèles d'AP, contenir des vidéos complètes pour utiliser pleinement toute l'information enregistrée par un TLI, et avoir des annotations très détaillées prenant en compte un grand nombre de phases de développement pour maximiser l'utilisation clinique potentielle.

Nous proposons ici une base de référence unique qui permettra à la communauté d'évaluer et de comparer les modèles MC et constituera une étape vers une FIV assistée par l'AP. Nos contributions sont illustrées en figure 5.1 et détaillées dans ce qui suit :

— Notre jeu de données contenant 704 vidéos complètes et un total de 2.4M d'images, suffisant pour entraîner et évaluer des modèles d'AP.
— Les annotations des paramètres MC, qui, contrairement aux travaux précédents, ne se limitent pas aux phases de division cellulaire précoce (t2-t5+), mais incluent aussi des divisions cellulaires tardives (t6 à t9+), des phases après la morulation (tM à tHB) et des phases très précoces (tPNa et tPNf). Ces phases n'ont été que peu utilisées jusqu'à présent.
— Des métriques d'évaluation adaptées au problème de l'extraction des paramètres MC.
— Des performance de références en utilisant les modèles de bases ResNet, LSTM et ResNet-3D.

5.0.1 Collection du jeu de données

Entre 2011 et 2021, 716 couples infertiles ont subi des cycles d'injection intracytoplasmique de spermatozoïdes ("Intracytroplasmic Sperm Injection", ICSI) dans le centre universitaire de FIV de Nantes et ont vu tous leurs embryons cultivés et suivis jusqu'au stade de blastocyste avec un système TLI. Seuls les cycles ICSI ont été inclus dans nos dispositifs time-lapse au cours de cette période, car les biologistes du laboratoire de FIV ont considéré que la FIV conventionnelle entraînerait des rythmes de développement différents de ceux de l'ICSI. L'éclosion assistée n'est pas systématiquement utilisée. Il n'y a pas eu de changements majeurs dans le laboratoire au cours de la période d'étude. Le traitement des patientes et le protocole de culture d'embryons ont été décrits dans une étude précédente [41]. La culture d'embryons a été réalisée depuis la fécondation (le premier jour) jusqu'au stade de blastocyste (jour 5 ou jour 6) à 37°C avec 5% O_2 et 6% CO_2 dans un milieu de culture séquentiel, c'est-à-dire *G1+* (Vitrolife©, Suède) du jour 1 au jour 3, suivi de *G2+* (Vitrolife©, Suède). Les images ont été acquises avec un système

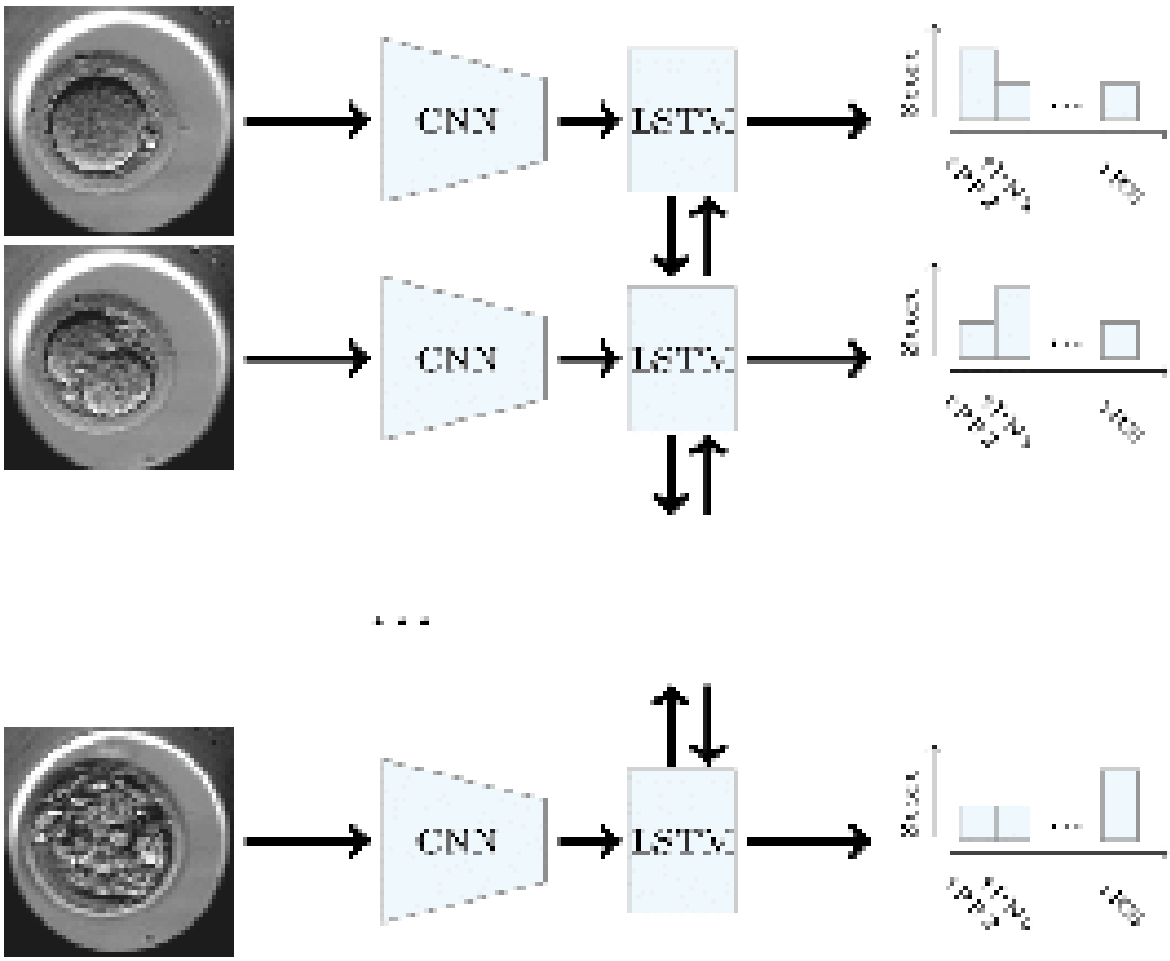


FIGURE 5.1 – Le jeu de données d'embryons time-lapse. Ce jeu de données contient 704 vidéos avec des annotations pour 16 événements morpho-cinétiques, accompagnées de 4 métriques d'évaluation et de 3 performances de modèles de base.

TLI (Embryoscope©, Vitrolife©, Suède) toutes les 10 à 20 min par une caméra sous une source lumineuse LED de 635 nm passant par une optique de modulation de contraste de Hoffman. Chaque vidéo a été annotée par un embryologiste qualifié et expérimenté soumis à un contrôle de qualité interne régulier.

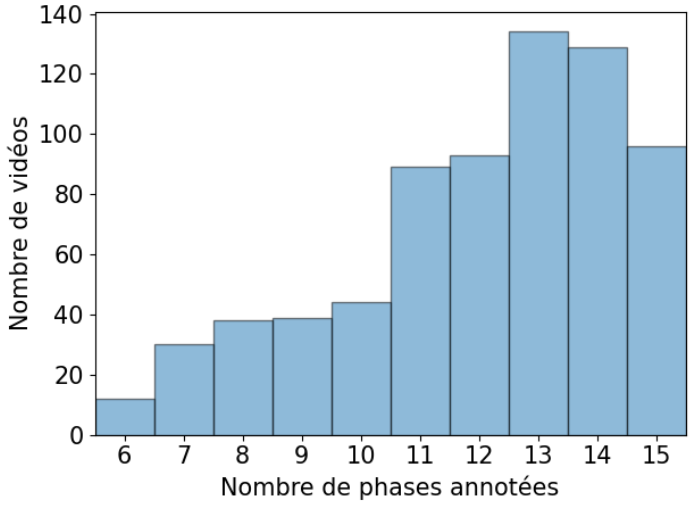
Sélection et caractérisation du sous-ensemble

Au cours de ces dix dernières années, le laboratoire de FIV du CHU de Nantes a au total enregistré plus de 22 000 vidéos d'embryons. Cependant, nous ne rendons publique qu'une partie de notre jeu de données avec les annotations des paramètres MC pour plusieurs raisons.

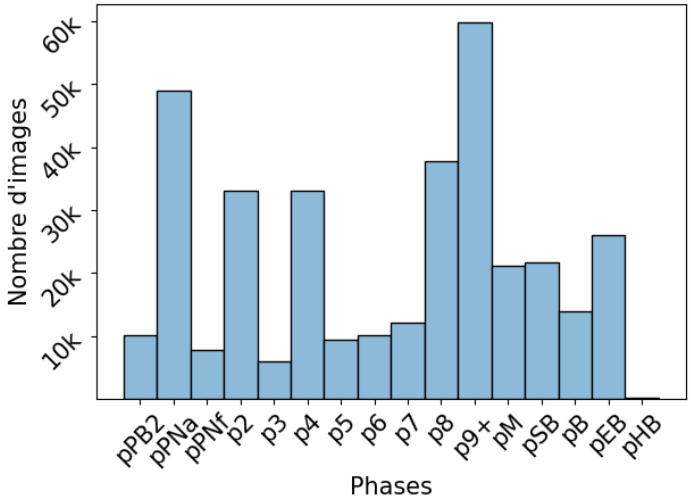
Premièrement, la tâche de reconnaissance des paramètres MC ne requiert pas la totalité du jeu de données que nous avons pour entraîner un bon modèle. Deuxièmement, comme vu précédemment, tout les paramètres MC ne sont pas annotés avec la même fréquence. En effet, le processus d'annotation prend du temps aux biologistes, c'est pourquoi seul les paramètres qu'ils jugent les plus importants sont souvent annotés. Le sous-ensemble de données présenté ici est constitué de 704 vidéos, ce qui correspond à une semaine d'extraction manuelle à partir de l'interface utilisateur du TLI, auxquelles on a retiré les vidéos comportant moins de 6 paramètres MC annotés, afin de ne garder que des vidéos avec une annotation détaillée. Nous avons choisi d'utiliser plus d'événements que la plupart des travaux précédents [87, 131, 80, 118] afin de développer des modèles qui peuvent décrire plus précisément le développement embryonnaire dans un environnement contrôlé. En conséquence, la plupart des vidéos ont au moins 8 phases annotées et environ 360 vidéos ont plus de 13 phases annotées, illustrant la richesse de l'annotation de notre jeu de données, comme le montre la figure 5.2a. De plus, chaque phase est représentée par au moins 10k d'images (cf. figure 5.2b), sauf la phase pHB qui n'est représentée que par un millier d'images, l'enregistrement de la vidéo étant souvent arrêté avant tHB.

Le problème d'apprentissage

On propose de voir le problème de reconnaissance automatique des paramètres MC comme une situation de classification d'images où un modèle va apprendre à distinguer les différents stades de développement de l'embryon. Pour cela il faut attribuer une étiquette parmi les 16 phases de développements de l'embryon à chaque image à partir des instants des événements. Le processus pour obtenir des étiquettes à partir des instants est décrit dans le paragraphe suivant. Notez que lors des expériences qui suivent, on utilise des modèles qui peuvent traiter une séquence d'images au lieu d'images isolées mais ces modèles doivent quand même prédire une étiquette pour chaque image de la séquence. La plupart des travaux précédents fusionnent ou ignorent des classes afin de réduire la variabilité des annotations. Au contraire, nous avons choisi d'utiliser les 16 phases disponibles car chacune d'entre elles peut potentiellement être pertinente pour l'évaluation de la qualité de l'embryon. Comme nous le verrons dans la tableau 5.2, les modèles atteignent de bonnes performances malgré la variabilité induite par l'utilisation de tous les stades.



(a) Distribution du nombre de phases par vidéos.



(b) Nombre d'images annotées pour chaque phase.

FIGURE 5.2 – Statistiques du jeu de données.

5.0.2 Des instants des événements aux étiquettes des images

Nous proposons de convertir les annotations des paramètres MC en étiquettes d'image afin de pouvoir les exploiter dans le cadre de la classification d'images. Cela signifie que nous devons attribuer une étiquette à chaque image et que le modèle sera entraîné à prédire les étiquettes. Cependant, les annotations données par les biologistes sont des instants, c-à-d. des mesures en heures post-fertilisation, qui indiquent la position temporelle des événements dans la vidéo.

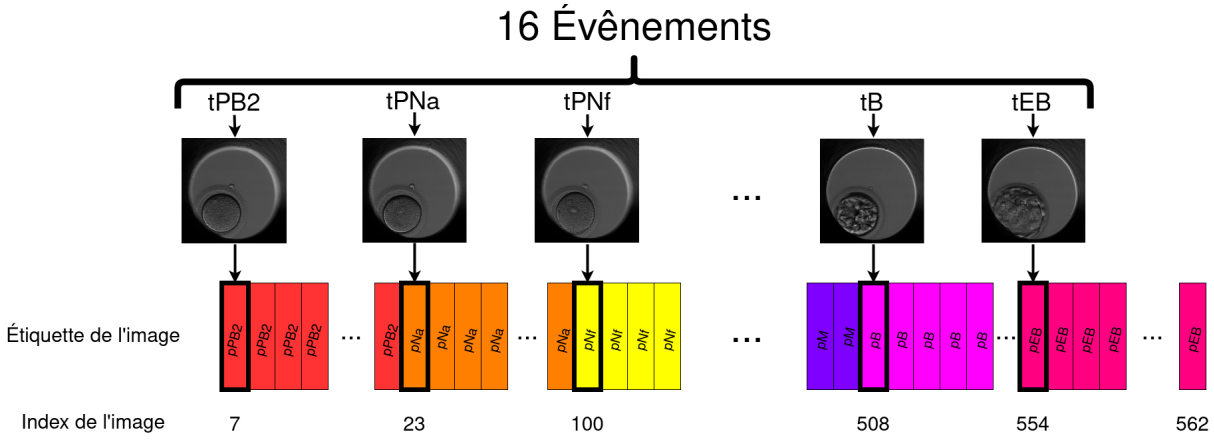


FIGURE 5.3 – La méthode utilisée consiste à attribuer une étiquette à chaque image de la vidéo. Tout d'abord, nous identifions à quelle image chaque événement se produit et nous attribuons à ces images une étiquette correspondant à l'événement qu'elles montrent. Les autres images reçoivent l'étiquette correspondant à l'événement le plus récent survenu dans les images précédentes. Notez que toutes les images sont étiquetées à l'exception de celles qui précèdent tPB2 car elles précèdent tous les événements.

Connaissant l'instant auquel chaque image a été prise, nous identifions les images correspondantes à chaque événement et nous leur attribuons une étiquette correspondant à l'événement qu'elles montrent (noté pPB2, pPNa, pPNf, p2, p3, p4, p5, p6, p7, p8, p9+, pM, pSB, pB, pEB ou pHB), comme illustré dans figure 5.3. Les autres images reçoivent l'étiquette correspondant à l'événement le plus récent survenu dans les images précédentes. Cet étiquetage construit la succession des phases de développement de l'embryon, délimitées par les événements cellulaires.

5.0.3 Des étiquettes d'images aux instants des événements

Une fois qu'un modèle a effectué une inférence sur toutes les images d'une vidéo, nous avons une séquence de sorties, où chaque sortie est une distribution sur les étiquettes possibles. Pour comparer les prédictions du modèle aux instants de la vérité terrain, nous devons convertir la séquence de sorties en une liste d'instants. La simple sélection de la phase ayant le score maximum à chaque image n'est pas une bonne solution car elle peut produire des séquences d'événements qui sont biologiquement impossibles. En effet, comme évoqué dans le chapitre précédent, les modèles utilisés n'ont pas de contrainte les obligeant à respecter la chronologie des phases de développement de l'embryon et cela peut conduire à des transitions en sens inverse durant les stades dont la définition est ambiguës (par exemple $p3 \rightarrow p2$, $pM \rightarrow p9+$, etc.). Par conséquent, nous proposons d'utiliser l'algorithme de Viterbi pour résoudre ce problème, comme

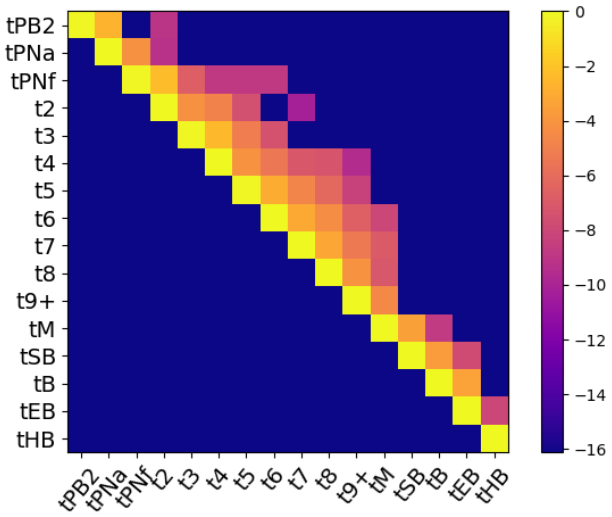


FIGURE 5.4 – Log-probabilités de transition entre les différents stades de développement de l'embryon. Notez que, pour cette visualisation, nous avons assigné une probabilité de $10^{-}6$ aux transitions qui n'arrivent jamais dans l'ensemble d'entraînement afin d'éviter des erreurs lors de l'application du logarithme.

cela est souvent fait dans la littérature [87, 80, 118, 109]. L'algorithme produit une séquence cohérente de prédictions en combinant la séquence de probabilités produite par le modèle avec une matrice de probabilité de transitions entre les différents stades de développement. Cette matrice est de taille 16×16 et indique à la ligne i et à la colonne j la probabilité que l'on assiste au stade j à la trame suivante sachant que la trame actuelle montre le stade i. Cette matrice est calculée empiriquement sur l'ensemble d'entraînement du modèle. Étant donné que les séquences d'événements biologiquement impossibles ne se produisent jamais dans l'ensemble d'apprentissage, leur probabilité est fixée à zéro et l'algorithme de Viterbi ne prédit jamais de telles séquences. La matrice de transition obtenue est visualisée en figure 5.4.

Enfin, nous construisons une liste d'instants en extrayant l'instant de la première image assignée à chaque étiquette, comme illustré en figure 5.5. Notez qu'un modèle peut parfois manquer un événement ou prédire un événement qui ne s'est pas produit. Nous expliquons dans la section 5.0.4 comment nous traitons ces cas au moment de l'évaluation.

5.0.4 Les métriques

Avec les données et les annotations, nous proposons plusieurs métriques pour évaluer les modèles sur ces données. La première métrique est la corrélation linéaire r entre les instants de

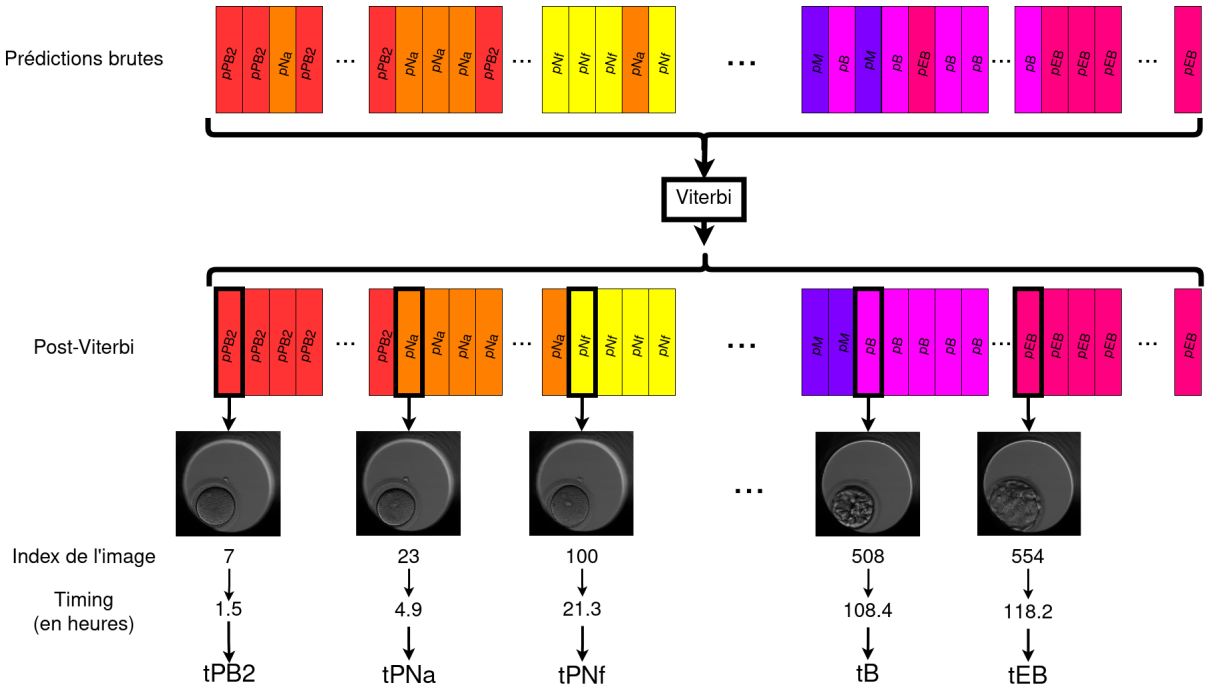


FIGURE 5.5 – La méthode utilisée pour obtenir un instant pour chaque événement à partir des prédictions du modèle.

transition prédits et les instants réel des transitions correspondantes. Avant d'être calculée, elle nécessite l'application préalable de l'algorithme de Viterbi afin que les prédictions des modèles soient rendues cohérentes tout au long de la vidéo. La corrélation r est calculée comme suit :

$$r = \frac{C}{V_p \times V_{gt}}, \tag{5.1}$$

où C, V_p et V_{gt} sont respectivement la covariance entre les instants de transition prédits et réels, la variance des instants prédits et la variance des instants réels. Étant donné qu'il peut arriver qu'un modèle manque un événement ou qu'un événement n'existe pas dans la vérité terrain, seules les transitions présentes à la fois dans la vérité du terrain et dans les prédictions sont prises en compte. Nous avons observé dans nos expériences que cette métrique est positivement biaisée, ce qui nous a conduit à introduire trois métriques supplémentaires : la précision p, la précision de Viterbi p_v et la précision temporelle p_t. La précision p est l'une des métriques les plus utilisées en classification d'images et est définie par la proportion d'images correctement étiquetées par le modèle :

$$p = \frac{N}{N_{total}}, \tag{5.2}$$

où N et N_{total} sont respectivement les nombres d'images correctement classées et le nombre total d'images. Nous définissons également une variante, la précision de Viterbi p_v qui consiste à calculer la précision une fois l'algorithme de Viterbi appliqué :

$$p_v = \frac{N_v}{N_{total}} \tag{5.3}$$

où N_v est le nombre d'images correctement classées une fois que les prédictions brutes ont été traitées à l'aide de l'algorithme de Viterbi.

Enfin, nous définissons la précision temporelle p_t comme la proportion moyenne de transitions de phase qui sont prédites suffisamment près de la transition réelle correspondante. Par "suffisamment proche", nous entendons que le temps séparant le moment de la transition prédite et le moment de la transition réelle est inférieur à un seuil. Cette métrique exige également que les prédictions soient rendues cohérentes en utilisant l'algorithme de Viterbi et est calculée comme suit :

$$p_t = \frac{T - T_{far}}{T}, \tag{5.4}$$

où T est le nombre total de transitions de phase et T_{far} est le nombre de transitions prédites trop éloignées dans le temps de leur moment réel.

Par exemple, considérons une vidéo contenant $T = 6$ transitions (p2 $\rightarrow$ p3 $\rightarrow$ p4 $\rightarrow$ p5 $\rightarrow$ p6 $\rightarrow$ p7 $\rightarrow$ p8) où le modèle a prédit la séquence (p2 $\rightarrow$ p3 $\rightarrow$ p4 $\rightarrow$ p5 $\rightarrow$ p6 $\rightarrow$ p8). Le modèle a omis la phase p7 mais a prédit la phase p8, ce qui est probablement dû à la longueur de la phase p7 qui est courte en général. Notez que le développement d'un embryon ne peut pas omettre une phase. En effet, lorsqu'une phase n'est pas visible dans la vidéo (et donc dans les annotations), c'est parce qu'elle s'est entièrement déroulée entre deux photographies successives, qui sont espacées de 20 à 30 minutes. En conséquence, lorsqu'un modèle omet une phase, nous considérons qu'il l'a implicitement prédite au même instant qu'il a prédit la phase suivante. Dans l'exemple précédent, nous considérons donc que le modèle a prédit la transition vers p7 au même moment que la transition vers p8. Maintenant, supposons que les transitions (p2 $\rightarrow$ p3) et (p3 $\rightarrow$ p4) sont prédites trop loin des transitions correspondantes, c'est-à-dire que la première image où le modèle a attribué l'étiquette de la nouvelle phase et l'image réelle correspondant à la nouvelle phase sont séparées par un intervalle de temps supérieur à un seuil θ. Nous avons alors $T_{far} = 2$ et la précision temporelle est $p_t = (6 - 2)/6 = 0.67$.

Le seuil θ doit être dépendant de la phase car certaines phases ont des définitions plus ambiguës que d'autres et sont donc plus difficiles à localiser précisément dans le temps que d'autres. Pour cela, nous utilisons les écarts types intra-opérateurs obtenus par Martínez-Granados

et al. pour avoir des seuils plus ou moins grands en fonction de l'ambiguïté intrinsèque de chaque phase [102]. Dans ce travail, les auteurs ont envoyé des vidéos time-lapse du développement embryonnaire à plusieurs centres de FIV comme programme de contrôle de qualité externe et ont notamment étudié la variance intra-opérateur. En utilisant leurs données supplémentaires, nous calculons l'écart-type σ_p observé entre opérateurs pour chaque phase p. Le seuil θ_p que nous utilisons pour la phase p est simplement fixé à σ_p.

Les écarts types pour chaque phase sont disponibles dans le tableau 5.1. Les métriques p et p_v ont l'inconvénient de pénaliser les modèles qui proposent des transitions de phase éloignées des transitions réelles autant que ceux qui se trompent de quelques images seulement. La métrique de la précision temporelle tient compte de ce fait : un modèle qui prédit un changement de phase proche du changement de phase réel est favorisé par rapport à un modèle qui est loin de la vérité.

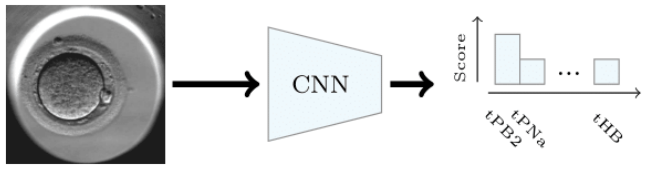
Phase	pPNa	pPNf	p2	$p3$	$p4$	$p5$	$p6$
σ_p	1.13	0.50	0.91	1.81	1.34	1.49	1.61
Phase	$p7$	$p8$	$p9+$	pM	pSB	pB	pEB
σ_p	2.93	5.36	4.42	5.46	3.78	3.29	4.85

TABLE 5.1 – Écart-type inter-opérateurs des annotations en heures. Calculé à partir des données de Martínez-Granados et al. [102].

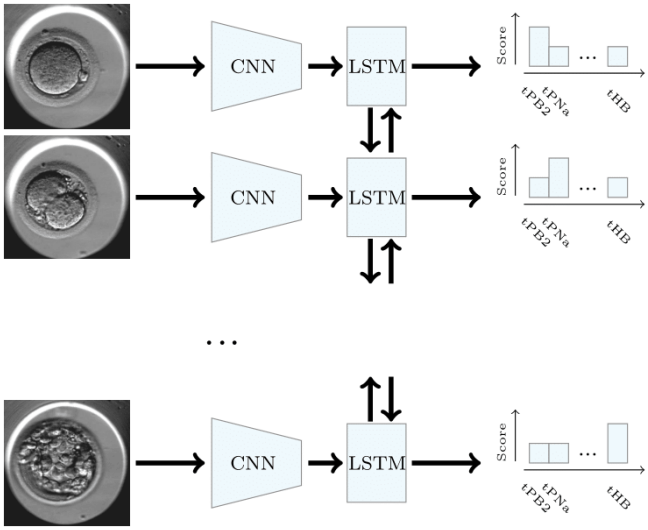
5.0.5 Les architectures

Plusieurs modèles de base ont été utilisés pour établir des performances de références. Le premier modèle est conçu pour la classification d'images isolées, les deux modèles suivants permettent la classification d'images dans une séquence. Par simplicité, les modèles présentés ici n'utilisent que les images du plan F0 car c'est le plan central et c'est donc les images de ce plan qui sont les plus nettes. Ils sont illustrés dans la figure 5.6 et détaillés ci-dessous.

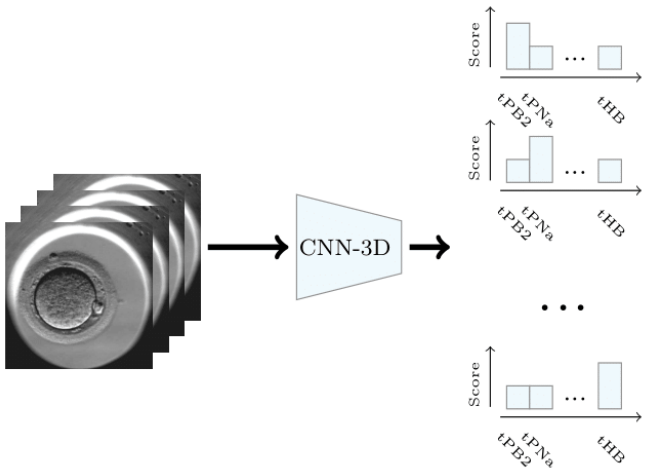
ResNet. Les modèles résiduels sont largement utilisés pour la classification d'images isolées, par exemple sur ImageNet [58]. Ce modèle est composé exclusivement de couches de convolution et contient des connexions résiduelles toutes les 2 couches. La résolution et le nombre de canaux des cartes de caractéristiques sont respectivement divisés et multipliés par 2 toutes les 4 couches. Après les convolutions, une couche de regroupement par moyenne produit un vecteur de caractéristiques, auquel la couche finale de soft-max est appliquée pour faire des prédictions. Nous utilisons la variante ResNet-18 proposée par He et. al. [58].



(a) ResNet



(b) ResNet-LSTM



(c) ResNet-3D

FIGURE 5.6 – Les différents architectures de bases évaluées. ResNet prend en entrée une image isolée et produit un vecteur de probabilités de classe. ResNet-LSTM et ResNet-3D ont en entrée une séquence d'images et produisent une séquence de vecteurs de probabilité.

ResNet-LSTM. Ce modèle est la combinaison du modèle ResNet avec un LSTM [61]. Le modèle LSTM a été conçu pour modéliser des séquences et a été appliqué avec succès dans des

tâches telles que la reconnaissance vocale [51]. Les pré-activations de l'avant-dernière couche de ResNet sont utilisées comme vecteur de caractéristiques et sont transmises à un LSTM bidirectionnel à deux couches qui modélise l'évolution temporelle des caractéristiques. La taille de chaque unité cachée est de 1024. Une couche linéaire après le LSTM calcule les scores de classe pour chaque image.

ResNet-3D. Le modèle ResNet-3D [55] est une variante de ResNet conçue pour la classification de séquences d'images. Ce modèle traite la séquence d'images en fusionnant les informations temporelles à toutes les couches du réseau, permettant ainsi une fusion à la fois tardive et précoce des informations. Afin que le modèle produise une prédiction par image de la séquence, les hyper-paramètres "max-pooling" et "stride" sont fixés à 1 dans la dimension temporelle. Nous utilisons la variante 'R2plus1d-18' proposée par Hara et al. [55].

5.0.6 Mise en place expérimentale

Nous utilisons deux GPU T4 avec PyTorch version 1.10.0.

Pré-traitement. La procédure de prétraitement que nous utilisons est largement inspirée de la procédure standard proposée par Krizhevsky et al. [85]. Pendant l'apprentissage, les images sont redimensionnées de 500×500 à 250×250 afin de réduire l'utilisation de la mémoire du GPU, puis une découpe aléatoire de taille 224×224 est extraite et enfin les images sont retournées verticalement avec une probabilité de 0,5 et retournées horizontalement avec une probabilité de 0,5. Lors de la validation et du test, les images sont également redimensionnées à 250×250, suivies d'un recadrage central de taille 224×224.

Hyper-paramètres. Chaque lot d'entraînement est composé de 10 séquences de 4 images consécutives. Le modèle ResNet traite chaque image indépendamment et lit donc les $10 \times 4 = 40$ images comme s'il s'agissait d'images indépendantes. Un nombre égal de séquences d'entrée et une longueur de séquence égale pour les trois modèles permettent une comparaison équitable. Nous utilisons une taille de lot de 10 car nous avons observé au cours de nos expériences qu'elle est suffisamment faible pour permettre un entraînement sur des séquences d'images et suffisamment grandes pour fournir une convergence suffisamment rapide pendant l'optimisation. La longueur de la séquence est fixée à 4 car c'est la longueur maximale possible compte tenu de la mémoire du GPU dont nous disposons. La position de la séquence dans la vidéo est choisie aléatoirement dans la vidéo. Nous utilisons la fonction d'entropie croisée standard,

optimisée avec SGD, avec un taux d'apprentissage constant et un momentum. Les valeurs du taux d'apprentissage et du momentum ont été fixées aux valeurs standards par défaut proposées par PyTorch, à savoir 0.001 et 0.9. Nous avons appliqué le dropout [137] sur la dernière couche de chaque modèle pendant l'apprentissage avec une probabilité $p = 0.50$ qui est également la valeur par défaut de PyTorch. Pendant le test et la validation, pour réduire l'utilisation de la mémoire du GPU, la taille du lot d'évaluation a été fixée à 150. Les modèles n'ont pas été évalués sur l'ensemble de la vidéo en une seule fois mais sur des séquences de 150 images. Comme chaque vidéo contient en moyenne environ 500 images, quelques inférences suffisent pour analyser une vidéo entière. Soit N le nombre total d'images d'entraînement et L le nombre d'images dans une séquence. Une époque se termine lorsque le modèle a vu N/L séquences. Pour sélectionner les séquences, nous avons utilisé un échantillonnage aléatoire uniforme avec remplacement, c'est-à-dire que le modèle peut voir la même image plusieurs fois et peut ne pas voir certaines images au cours d'une époque. Pour chaque division, nous avons utilisé 564, 70 et 70 vidéos pour l'entraînement, la validation et le test. Cela représente respectivement 80%, 10% et 10% des vidéos. Afin d'obtenir des mesures de performances plus fiables, nous réalisons une validation croisée avec $k = 5$. Un modèle est entraîné pendant 10 époques. Le meilleur modèle considérant l'ensemble de validation est ensuite restauré et évalué sur l'ensemble de test.

Initialisation des poids. Les poids de ResNet et ResNet-3D sont pré-entraînés sur ImageNet [85] et Kinetics [55] respectivement. Les poids de la composante ResNet de ResNet-LSTM sont également pré-entraînés sur ImageNet et les poids des composantes LSTM sont initialisés de manière aléatoire.

5.0.7 Résultats

Les performances des architectures ResNet, ResNet-LSTM et ResNet-3D sont compilées dans le tableau 5.2.

Modèle	Traitement des images	r	p	p_v	p_t
ResNet	Isolé	0.961 ± 0.026	0.663 ± 0.041	0.701 ± 0.044	0.371 ± 0.09
ResNet-LSTM	En séquence	$\mathbf{0.977 \pm 0.009}$	0.685 ± 0.041	0.696 ± 0.043	0.559 ± 0.223
ResNet-3D	En séquence	0.97 ± 0.021	$\mathbf{0.705 \pm 0.036}$	$\mathbf{0.735 \pm 0.042}$	$\mathbf{0.659 \pm 0.154}$

TABLE 5.2 – Performance des modèles de base obtenues. r est la corrélation, p est la précision, p_v est la précision de Viterbi et p_t est la précision temporelle.

La première métrique que nous avons considérée est la métrique de corrélation. qui se montre ici peu informative. En effet, elle a un biais élevé et une faible variance et les valeurs obtenues sont toutes proches de 1. Cela est dû au fait que les transitions prédites sont forcées d'être dans un ordre biologiquement plausible après l'application de l'algorithme de Viterbi, ce qui implique un niveau minimum d'alignement avec les transitions réelles. Pour avoir une meilleure idée des performances des modèles, nous avons concentré notre analyse sur d'autres métriques. La précision p fournit des valeurs plus espacées et montre que ResNet, ResNet-LSTM et ResNet-3D sont respectivement capables de classer correctement en moyenne 0.66, 0.68 et 0.70 des images des vidéos de test. Logiquement, la précision avec Viterbi p_v donne des valeurs plus élevées que la précision régulière p car les prédictions du modèle sont d'abord rendues biologiquement plausibles. Enfin, la précision temporelle montre que ResNet, ResNet-LSTM et ResNet-3D prédisent respectivement 0.37, 0.55 et 0.65 des transitions à un instant proche de la réalité. Ces 3 métriques soulignent la supériorité de ResNet-LSTM et ResNet-3D sur ResNet. Cela s'explique par le fait que ResNet traite les images de manière isolée, comme un embryologiste ayant une vue statique de l'embryon à l'aide d'un microscope, alors que ResNet-LSTM et ResNet-3D traitent plusieurs images ensemble, comme un embryologiste utilisant un système TLI, et bénéficient donc de contexte pour mieux comprendre à quelle phase de développement se trouve l'embryon. Étant donné que les modèles de base ResNet-LSTM et ResNet-3D conçus pour le traitement des vidéos sont plus performants que le modèle conçu pour le traitement des images ResNet, cela souligne la pertinence de proposer un jeu de données composé de vidéos complètes plutôt que d'images isolées. De plus, les modèles de base ont obtenu de bonnes performances, ce qui démontre que notre jeu de données est suffisant en taille et en qualité pour entraîner et évaluer les modèles d'AP.

Évaluation de la difficulté du jeu de donnée. Pour évaluer la difficulté de ce jeu de données, il n'est pas possible de comparer directement les performances de base obtenues ici avec les performances de base des travaux précédents car ils utilisent des ensembles de classes différents. Cependant, étant donné que les travaux précédents utilisent des ensembles de classes restreints par rapport à l'ensemble utilisé dans ce travail, nous proposons de réévaluer les modèles de base tout en ignorant et en fusionnant certaines classes pour obtenir un ensemble de classes qui est similaire à ceux couramment utilisés dans la littérature. Plus précisément, nous avons d'abord évalué la capacité des modèles à identifier les phases de divisions cellulaires précoces comme cela est souvent fait dans la littérature [80, 87, 97, 109, 118, 131] et ensuite nous avons évalué leur capacité à discriminer entre blastocyste et non-blastocyste comme proposé par Kanakasabapathy

et al. [78].

Pour reproduire l'ensemble de classes de divisions cellulaires, nous avons supprimé les images de test appartenant aux phases avant p2 et après p9+ et fusionné les classes de p5 à p9+ en une seule classe appelée p5+. En utilisant cette configuration, nous avons obtenu des précisions similaires à celles trouvées dans la littérature : 0.86, 0.88 et 0.88 pour ResNet, ResNet-LSTM et ResNet-3D, contre 0.82 à 0.87 dans la littérature [80, 87, 97, 109, 118, 131] (cf. tableau 5.3). Pour tester la performance de l'identification des blastocystes, nous avons re-traité les prédictions faites lors de la première évaluation et fusionné les phases de tPB2 à tM pour la classe des non-blastocystes et fusionné les phases de tB jusqu'à la fin pour la classe des blastocystes. Nous avons ignoré la phase pSB car il s'agit d'une phase de transition vers le stade blastocyste n'appartenant à aucun des 2 groupes. Nous avons obtenu des précisions de 0.98, 0.99 et 0.99 contre 0.96 dans la littérature [78] (tableau 5.3). Étant donné que la performance de base sur ce jeu de données est proche de la performance de base trouvée dans les travaux précédents, nous pouvons conclure que notre base de données est similaire en difficulté et en qualité aux jeux de données précédents.

Modèle	Identification des phases de p2 à p5+	Blastocyste vs Non blastocyste
ResNet	0.86	0.98
ResNet-LSTM	0.88	0.99
ResNet-3D	0.88	0.99

TABLE 5.3 – Évaluation des architectures de base sur l'identification des phases de p2 à p5+ et blastocyste vs non-blastocyste. La métrique utilisée est la précision p.

Comparaison avec une approche *ad-hoc* Nous avons ensuite cherché à comparer les performances obtenues avec les méthodes d'AP avec la méthode *ad-hoc* mentionnée dans le chapitre précédent [39]. Notons que nous n'avons pas utilisé la métrique p_v car elle nécessite d'appliquer l'algorithme de Viterbi, ce qui n'est pas possible pour la méthode *ad-hoc*, car elle ne génère pas de probabilités de transition mais prédit directement les instants des transitions. Les métriques p, p_v et p_t sont calculées dans le tableau 5.4 et montrent que les modèles d'AP surpassent largement la méthode *ad-hoc* (0.659 vs. 0.615 sur la métrique pt et 0.705 vs. 0.58 sur la métrique p) confirmant l'intérêt de l'AP pour la tâche d'extraction automatique des paramètres MC (tableau 5.4).

Modèle	r	p	p_v	p_t
ResNet	0.961	0.663	0.701	0.371
ResNet-LSTM	**0.977**	0.685	0.696	0.559
ResNet-3D	0.97	**0.705**	**0.735**	**0.659**
Ad-hoc [39]	0.973	0.580	0.580	0.615

TABLE 5.4 – Performances des méthodes d'AP par rapport au modèle *ad-hoc* proposé par Feyeux et al. [39]. Les lignes du haut indiquent les performances moyennes des modèles d'AP, la ligne du bas indique les performances moyennes de la méthode *ad-hoc* sur l'ensemble du jeu de données. Pour chaque métrique, les caractères gras indiquent la meilleure performance moyenne pour la métrique donnée.

5.0.8 Discussion

Dans cette étude, nous proposons un jeu de données de vidéos de développement embryonnaire en time-lapse et le rendons accessible au public afin de faciliter et d'améliorer les recherches futures dans ce domaine. Ce jeu de données est accompagné d'annotations MC détaillées et de métriques adaptées au problème. Nous avons également constaté que des modèles de base simples peuvent être entraînés avec de bonnes performances, ce qui montre que le jeu de données est suffisamment grand pour entraîner un modèle d'AP.

Nous avons montré que les approches d'AP permettent de surpasser les performances obtenues précédemment avec une méthode *ad-hoc*. De plus, en exploitant des modèles de séquence d'images comme ResNet-3D ou ResNet-LSTM, nous avons pu améliorer la qualité des prédictions, ce qui démontre la pertinence de proposer des vidéos complètes plutôt que des images isolées.

Les bonnes performances peuvent paraître surprenantes car ce jeu de données est composé de seulement 704 vidéos et la classification de vidéos peut être considérée comme requérant plus en données que la classification d'images. Cependant, la classification vidéo consiste à transmettre une séquence d'images à un modèle et à l'entraîner à produire une sortie unique où chaque vidéo a une étiquette unique pour toutes ses images. Ici, les modèles sont également transmis sous la forme d'une séquence d'images, mais ils sont entraînés à produire une sortie par image, c'est-à-dire à classifier chaque image, et chaque image a sa propre étiquette. C'est pourquoi nous considérons ce problème comme une tâche de classification d'images. La taille de ce jeu de données (342k images) est cohérente avec la taille des jeux de données trouvés dans la littérature, où le nombre d'images varie de 60k à 4M (cf. section 4.2.3).

Kanakasabapathy et al. ont signalé que la variance inter et intra était trop élevée lorsque plus

de 6 phases de développement de l'embryon étaient utilisées [78]. Au contraire, nous rapportons ici l'analyse de vidéos comprenant 16 phases de développement précises. Bien qu'une certaine variance soit également constatée dans notre travail, nous avons pu reconstituer avec précision la succession des 16 événements MC. Notre travail va donc au-delà la plupart des travaux précédents, qui ne prennent en compte que les phases précoces jusqu'à p9+ et en fusionnant les phases p4 à p9+ en une seule classe p4+. Nous avons également montré que ce jeu de données est similaire en difficulté aux jeux de données précédents en réévaluant les modèles de base en utilisant des ensembles de classes habituellement utilisés dans les travaux précédents.

Une autre partie intéressante de notre travail est que nous avons implémenté deux améliorations méthodologiques. Tout d'abord, nous avons effectué une validation croisée, alors que les études précédentes n'en n'ont pas utilisé. Deuxièmement, nous avons utilisé une architecture CNN 3D, ce qui est pertinent compte tenu de la nature temporelle des données et a permis d'améliorer les performances. Bien qu'elle n'ait pas été évaluée jusqu'à présent dans le domaine de la FIV et des vidéos en time-lapse du développement embryonnaire, l'architecture ResNet-3D a été utilisée avec succès dans plusieurs autres domaines médicaux tels que l'oncologie [167, 126], la cardiologie [57], la qualité de l'imagerie par tomodensitométrie (TDM), [24] et la neuroimagerie [129, 54].

Nous avons mentionné au début de ce chapitre que cette base de référence est un échantillon de la base de données recueillie et annotée par le laboratoire de FIV du CHU de Nantes, qui contient au total plus de 22000 vidéos. Également, d'autres type d'annotations sont disponibles comme des notes attribués par les biologistes aux embryons [149] et l'issue de la procédure FIV, qui indique si un embryon a mené ou non à une grossesse par exemple. Exploiter l'ensemble des données et annotations disponibles pourrait probablement permettre d'améliorer les performances obtenues mais aussi d'obtenir une évaluation plus robuste. Il serait également possible d'évaluer les liens qui existent entre les variables représentées par les différents types d'annotation du point de vue de l'impact sur l'apprentissage du modèle. En effet, la littérature a par exemple montré qu'il existe un lien entre les paramètres MC et l'issue de la FIV [117, 112, 8]. Il serait donc intéressant d'examiner si pré-entraîner un modèle à identifier les stades de développement faciliterait la généralisation sur la tâche de prédiction de l'issue de la FIV.

À retenir

— Le jeu de données présenté ici contient plus de 700 vidéos 3D montrant chacune le développement d'un embryon en accéléré.
— La quantité de données de cette base publique est suffisante pour entraîner des modèles d'AP.
— Cette base propose suffisamment de contexte temporel pour pouvoir l'exploiter avec des modèles conçus pour le format vidéo.
— Contrairement à la plupart des travaux précédent, les annotations détaillent le développement de l'embryon à travers 16 stades différents, permettant aux modèles de fournir des prédictions riches.

CHAPITRE 6

COMPARAISON DE LA FIABILITÉ DES MODÈLES D'ATTENTION ET DES MÉTHODES D'EXPLICATION POST-HOC APPLIQUÉS AUX VIDÉOS EMBRYONNAIRES

Dans ce chapitre, nous comparons la fiabilité des cartes de saillance générées par des modèles d'attention (dont le modèle BR-NPA présenté précédemment) et des méthodes d'explications post-hoc appliquées au problème de l'identification des stades embryonnaires.

6.1 Introduction

Les incubateurs à imagerie time-lapse (TLI) semblent être la solution la plus prometteuse pour améliorer les méthodes d'évaluation de la qualité des embryons et, par la suite, l'efficacité clinique de la FIV. En particulier, le volume élevé sans précédent d'images de haute qualité produites par les systèmes TLI a déjà été exploité à l'aide de méthodes d'apprentissage profond (DL). Les applications précédentes de DNN aux images de systèmes time-lapse ont notamment porté sur la conception de modèles permettant d'identifier automatiquement les stades de développement de l'embryon, une tâche cruciale pour identifier les embryons à faible potentiel de grossesse [80, 118]. Cependant, une limite importante au développement de solutions basées sur l'IA pour la FIV est la nature de boîte noire de la plupart des modèles de pointe, en raison de la complexité des architectures d'AP, qui soulève des problèmes potentiels de partialité et d'équité [4]. Le besoin d'une IA interprétable a augmenté non seulement dans le domaine de l'IVF mais aussi dans la communauté de l'apprentissage profond en général. Cela a donné naissance à une tendance dans la littérature où les auteurs confrontent les utilisateurs aux décisions des modèles accompagnées de diverses explications pour étudier la perception des explications

par les utilisateur pour une application particulière [6, 144, 147]. Cependant, le coût financier et la difficulté d'établir un protocole correct rendent cette approche difficile à mettre en place. En raison de ces problèmes, une autre tendance propose d'étudier la fiabilité des cartes de saillance produites par les diverses méthodes d'explications proposées dans la littérature à l'aide des métriques de fiabilité présentées en section 1.3 du chapitre 1. Dans ce chapitre, nous nous inscrivons dans cette tendance, et comparons la fiabilité de modèles d'attention et de méthodes post-hoc appliquées au problème de l'identification du stade embryonnaire à l'aide de ces métriques dédiées. Tout d'abord, nous décrivons les modèles d'attention, les méthodes post-hoc et les mesures de fidélité que nous utilisons dans ce chapitre. Nous comparons différentes approches en utilisant des métriques de fiabilité et nous montrons empiriquement que (1) selon le type de métrique utilisée, les méthodes post-hoc ou les modèles d'attention sont favorisés et (2) parmi les approches étudiées, BR-NPA et Score-CAM fournissent la meilleure fiabilité. Ensuite, une étude qualitative montre que les cartes de saillance de BR-NPA permettent de comprendre quels sont les indices utilisés par le modèle (en l'occurrence, des indices pertinents du point de vue biologique) alors que les cartes de Score-CAM sont ambiguës et ne permettent pas une telle conclusion, à cause de leur faible résolution. Nous concluons par des remarques générales sur la nécessité de repenser ces métriques de fiabilité et de comprendre la relation entre le type de métrique et le type d'approche qui est favorisée.

6.2 Méthode

Nous utilisons les métriques DAUC, IAUC, IIC, AD et ADD présentées précédemment en chapitre 1 ainsi que les métriques DC et IC présentées en chapitre 2. Les approches utilisées pour générer des cartes de saillances sont listées dans la sous-section suivante.

6.2.1 Les approches pour générer des cartes de saillances

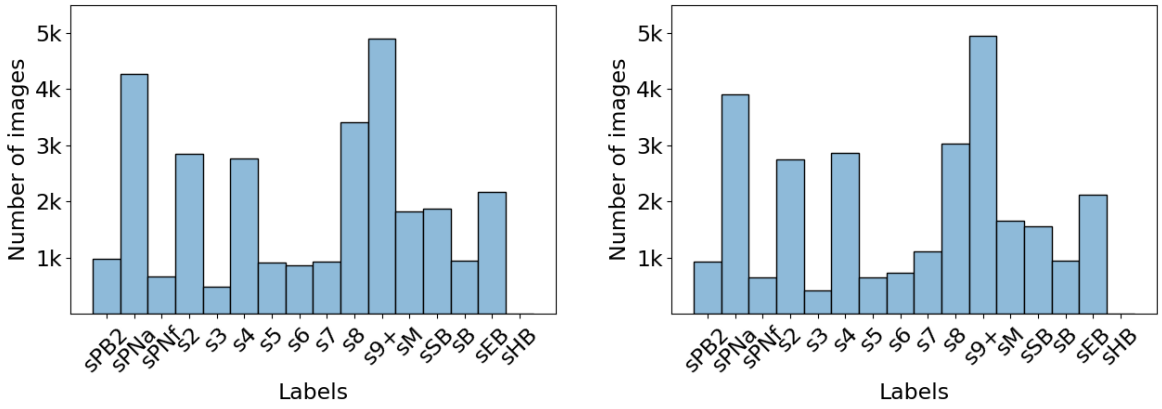
Nous décrivons maintenant les modèles et les méthodes utilisés pour générer des cartes de saillance. Tout d'abord, nous énumérons plusieurs modèles d'attention car ils intègrent le calcul d'une carte de saillance (appelée carte d'attention) qui est utilisée pour guider le processus de décision. Ensuite, nous énumérons des méthodes génériques d'explication post-hoc qui peuvent générer des cartes de saillance pour un large éventail de modèles et d'architectures sans nécessiter d'entraînement du modèle.

Modèles d'attention. Dans cette catégorie, nous incluons les modèles comprenant une couche d'attention spatiale dans leur phase d'inférence. Nous avons évoqué dans le chapitre 1 que les modèles d'attention de la littérature se distinguent en approches convolutionelles, prototypiques et non-paramétriques. Nous incluons donc dans notre étude un modèle de chaque catégorie, à savoir B-CNN [64], IBP [69] et BR-NPA, présentés précédemment. Enfin, nous utilisons l'architecture ABN car elle a déjà été appliqué au problème de la prédiction de la qualité des embryons [124], un problème également basé sur les vidéos time-lapse. Notez que nous n'incluons pas ProtoPNet et ProtoTree, des modèles que nous avons pourtant utilisés au chapitre 3, car nous n'avons pas réussi à entraîner ces modèles sur les données embryonnaire avec le code donné par les auteurs.

Méthode d'explication post-hoc. Cette catégorie propose des méthodes génériques qui peuvent être appliquées à n'importe quel CNN et ne nécessite pas d'entraînement du modèle, contrairement aux modules d'attention mentionnés ci-dessus. Parmi les méthodes présentées précédemment, nous incluons d'abord Grad-CAM pour sa popularité. Ensuite, nous ajoutons Grad-CAM++, Score-CAM et Ablation-CAM en tant que méthodes qui améliorent et supplantent Grad-CAM. Enfin, nous terminons avec RISE, une méthode avec une approche conceptuelle différente de Grad-CAM. Nous n'utilisons pas les méthodes d'explications SmoothGrad, VarGrad et GP au vu de leurs faibles résultats observés lors de l'étude qualitative en section 3.2.2 du chapitre 3.

6.2.2 Le jeu de données de vidéos d'embryons

Nous utilisons le sous-ensemble de 704 vidéos embryonnaires introduit au chapitre 5 pour la prédiction des paramètres MC. L'ensemble de données brutes se compose de 704 vidéos que nous avons divisées de manière égale en un ensemble d'entraînement/validation et un ensemble de test. Étant donné que nous nous concentrons sur la classification d'image, le jeu de données de vidéo est convertie en jeu de données d'images. Pour chaque ensemble et chaque vidéo, nous extrayons un tiers des images régulièrement espacées dans la vidéo. La taille totale de chaque ensemble est de 29843 images pour l'ensemble d'entraînement/validation et de 28282 images pour l'ensemble de test. Les modèles étudiés dans ce chapitre sont entraînés à traiter une image d'embryon et à prédire à quel stade de développement se trouve l'embryon. La distribution des classes est visible en figure 6.1.



(a) Training/validation set

(b) Test set

FIGURE 6.1 – La distribution des classes de l'ensemble de données utilisé. Il n'y a que 5 et 15 échantillons de pHB respectivement dans l'ensemble d'entraînement/validation et l'ensemble de test, car cette étape se produit généralement après l'arrêt de l'enregistrement vidéo.

6.3 Détails d'implémentations

La colonne vertébrale utilisée pour tous les réseaux est ResNet-50 [58]. Les images sont augmentées pendant l'entraînement en utilisant un recadrage aléatoire de 448×448 et un retournement horizontal aléatoire. Pendant le test, nous extrayons un recadrage central de taille 448×448. Nous utilisons 10% des images de l'ensemble d'apprentissage pour la validation. Les modèles sont entraînés en utilisant l'entropie croisée pendant 10 époques. Le meilleur modèle sur l'ensemble de validation est restauré pour la phase de test. Les hyperparamètres suivants ont été recherchés sur l'ensemble de validation en utilisant la bibliothèque python Optuna [5] avec l'échantillonneur par défaut (un algorithme de Parzen Estimator structuré en arbre) : le taux d'apprentissage, le *momentum*, l'optimiseur, la taille du lot, le *dropout* sur la couche de classification et le *weight decay*. Nous utilisons Pytorch 1.10.2 [26] et deux GPU P100. En suivant le travail original de Petsiuk et al. [115], nous avons échantillonné 4000 masques à une résolution de 7×7 pour la méthode post-hoc RISE.

6.4 Résultats

Le tableau 6.1 montre les résultats de l'évaluation de la fidélité des méthodes d'explication et des modèles d'attention sur l'ensemble de test. Étant donné la taille de l'ensemble de test et le temps de calcul conséquent des métriques et de certaines méthodes post-hoc, nous

échantillonnons aléatoirement 100 images de l'ensemble de test et calculons la performance moyenne de chaque approche pour chaque métrique sur ces images. Notez que nous incluons également la précision moyenne par vidéo des modèles sur l'ensemble de test pour souligner que tous les modèles ont des niveaux de précisions similaires. Cela signifie que les différences de fidélité observées ne peuvent pas être expliquées par des niveaux de précision différents mais uniquement par l'approche (modèle d'attention/méthode post-hoc) ou les métriques. Les résultats du tableau 6.1 montrent que les métriques de fidélité ne s'accordent pas sur le meilleur modèle. Par exemple, selon la métrique, la méthode d'explication la plus fidèle est BR-NPA, RISE, Ablation-CAM, ABN, ou Score-CAM et la moins fidèle est ABN, InterByParts, RISE, BR-NPA, ou AM. Dans le paragraphe suivant nous proposons d'étudier plus en détail le désaccord entre les métriques à l'aide de la corrélation τ de Kendall [79].

Modèle	Méthode de vis.	DAUC↓	IAUC↑	DC↑	IC↑	IIC↑	AD↓	ADD↑	Précision↑
CNN	AM	0.134	0.308	−0.174	0.075	0.19	0.397	0.325	71.0
	Grad-CAM++	0.1162	0.333	−0.101	0.032	0.52	0.14	0.383	
	RISE	0.113	**0.457**	−0.221	−0.077	0.5	0.137	0.436	
	Score-CAM	0.1079	0.315	−0.123	0.081	0.52	**0.108**	0.362	
	Ablation-CAM	0.0954	0.329	**0.272**	−0.071	**0.57**	0.111	0.328	
ABN	-	0.1464	0.249	−0.186	**0.136**	0.12	0.591	0.475	71.0
InterByParts	-	0.0876	0.115	0.196	−0.255	0.08	0.901	0.879	**71.3**
B-CNN	-	0.0772	0.221	−0.208	0.124	0.13	0.491	0.482	70.2
BR-NPA	-	**0.0709**	0.261	0.185	−0.146	0.04	0.91	**0.887**	70.7

TABLE 6.1 – Fiabilité et performance des approches étudiées.

Quantification du désaccord entre les métriques. Nous proposons d'utiliser le coefficient de corrélation τ de Kendall [79] pour avoir une meilleure vision du désaccord entre les métriques. Ce coefficient de corrélation indique si deux variables ont une relation monotone l'une avec l'autre. Lorsqu'il est proche de 1/-1, les variables ont une relation croissante/décroissante, et lorsqu'il est proche de 0, elles sont indépendantes. Nous représentons chaque métrique à l'aide des 9 performances moyennes qu'elles ont attribuées à chaque approche. Soit x et y les scores donnés par deux métriques aux 9 méthodes d'explications étudiées., i.e. deux colonnes du tableau 6.1. Nous définissons une paire concordante par une paire d'explications A et B qui sont classées de la même façon dans x et y, c'est-à-dire que les métriques x et y s'accordent pour dire que A est meilleure ou pire que B. Nous définissons également une paire discordante, où x et y sont en désaccord sur le classement de A et B. En outre, nous disons qu'une paire est ex aequo selon x si

A et B ont le même classement en x. Notez que cela implique que A et B aient exactement le même score selon la métrique x, ce qui est très peu probable étant donné que les scores sont des nombres réels. Enfin, le coefficient de corrélation τ de Kendall entre deux métriques x et y est calculé comme suit :

$$\tau(x, y) = \frac{(P - Q)}{\sqrt{P + Q + U} * \sqrt{P + Q + T}}, \tag{6.1}$$

où P est le nombre de paires concordantes, Q le nombre de paires discordantes, T le nombre d'égalités uniquement dans x, et U le nombre d'égalités uniquement dans y. Si une égalité se produit pour la même paire à la fois dans x et y, elle n'est ajoutée ni à T ni à U. Les valeurs des coefficients obtenus entre chaque paire de métriques sont représentées en figure 6.2. Globalement, la valeur de corrélation entre les métriques est faible. Cela signifie que les métriques sont globalement en désaccord sur le classement des modèles, de nombreuses métriques produisant un classement indépendant (DC et IC ont une corrélation proche de 0 avec AD, ADD, IIC et IAUC) et certaines métriques classant même les modèles dans un ordre proche de l'ordre inverse (AD et ADD, ADD et IIC, IC et DC). Cependant, certaines métriques offrent des classements similaires, comme AD avec IIC ou IAUC. Afin de mieux identifier les métriques se comportant de façon similaire, nous proposons une visualisation des classements données par les métriques à chacune des 100 images dans le paragraphe suivant.

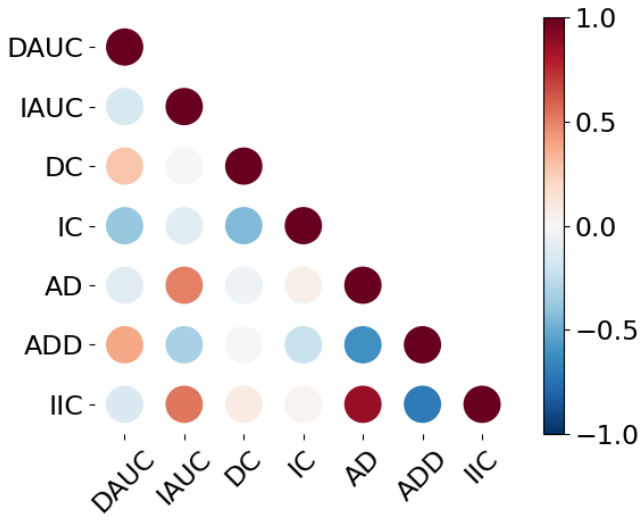


FIGURE 6.2 – Coefficient τ de corrélation de Kendall entre les métriques.

Visualisation des similarités entre les métriques. Pour obtenir une visualisation des similarités des métriques entre elles, nous proposons d'utiliser un algorithme de réduction de dimension tel que t-SNE [100]. Pour cela, nous avons repris les 9 valeurs données par chacune des métriques sur chacune des 100 images de test utilisée pour le tableau 6.1. Ces données sont ensuite interprétées comme des vecteurs en dimension 9 et projetée dans un espace en dimension 2 avec l'algorithme de réduction de dimension t-SNE. Notez que nous avons exclu la métrique IIC car il s'agit d'une métrique qui a une valeur binaire lorsqu'elle est calculée sur une seule image. Nous obtenons 6 nuages de points dans un espace à 9 dimensions : chaque nuage est composé de 100 points et représente une métrique, et chaque point représente les scores donnés par une métrique sur une image. Au total, on a donc $6 \times 100 = 600$ points. Nous proposons d'utiliser t-SNE pour réduire la dimension des 600 points en même temps et visualiser le résultat sur un seul graphique 2D. Notez qu'une fonction de distance appropriée doit être choisie, étant donné que chaque point représente un classement. Au lieu de la distance euclidienne par défaut, nous utilisons la fonction de distance suivante basée sur la corrélation τ de Kendall qui permet de comparer des classements :

$$D(x, y) = -\log(\frac{1}{2}(\tau(x, y) + 1)), \tag{6.2}$$

où $\tau(\cdot, \cdot)$ est la corrélation τ de Kendall. Avec cette métrique, nous assurons une distance qui va de 0 à $+\infty$ lorsque la corrélation $\tau(x, y)$ va de 1 à -1. Afin d'éviter une erreur de calcul numérique, lorsque $\tau = -1$, sa valeur est remplacée par $-0,999$. Il faut noter que nous n'avons pas utilisé l'algorithme UMAP plus récent [103] car la seule implémentation UMAP disponible nécessite la compilation de la fonction de distance personnalisée avec Numba, ce qui n'est actuellement pas possible avec l'implémentation du τ de Kendall de la librairie python Scikit-Learn [113].

La figure 6.3 montre que les métriques se structurent en deux groupes : {DAUC,DC,ADD} et {IAUC,IC,AD}. Étant donné que les métriques DAUC, DC et ADD consistent à masquer les zones importantes de l'image, nous appelons "*Mask*" ("Masquer") ce groupe de métriques. De même, les métriques IAUC, IC et AD consistent à mettre en valeur les zones importantes de l'image, nous appelons donc ce groupe "*Highlight*" ("Mettre en valeur"). C'est pourquoi nous appelons *Mask* ("Masquer") et *Highlight* ("Mettre en valeur") les deux groupes de métriques obtenus : {DAUC, DC, ADD} et {IAUC, IC, AD, IIC}. Notez que nous incluons également la métrique IIC dans le groupe *Highlight* car elle consiste également à mettre en évidence les zones saillantes de l'image. Cela démontre que la méthode utilisée pour mesurer l'impact d'une zone (mise en évidence ou masque) de l'image a un impact majeur sur le classement produit. Dans la suite, nous proposons d'agréger les classements produits par les métriques de chaque groupe

pour identifier les approches produisant les explications les plus fidèles.

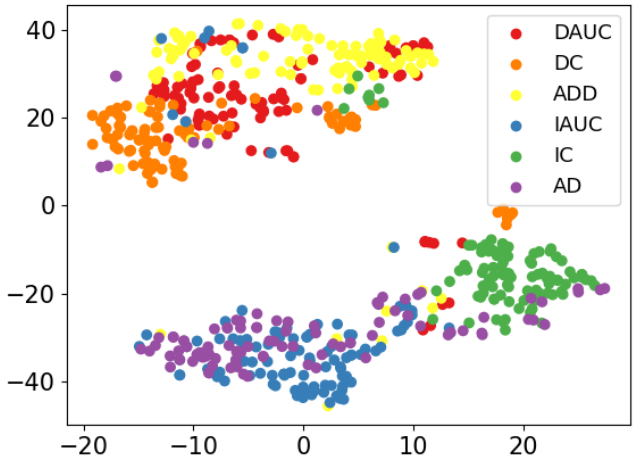


FIGURE 6.3 – Projection t-SNE du classement donné par les métriques sur les 100 images. La distance entre les points est calculée en utilisant une fonction basée sur la corrélation τ de Kendall [79].

Identification des meilleures approches. Nous proposons maintenant d'identifier les meilleures approches en fonction de chaque groupe de métriques. Pour cela, nous calculons le rang moyen d'une approche A sur le groupe *Highlight* et *Mask* comme suit :

$$\begin{aligned} r_{Mask}(A) &= \frac{r_{DC}(A) + r_{DAUC}(A) + r_{ADD}(A)}{3} \\ r_{Highlight}(A) &= \frac{r_{IC}(A) + r_{IAUC}(A) + r_{AD}(A) + r_{IIC}(A)}{4}, \end{aligned} \tag{6.3}$$

où $r_X(A)$ est le rang de A selon la métrique X. Le tableau 6.2 montre les classements moyens calculés. Le groupe de métrique de type *Mask* semble favoriser les modèles d'attention tandis que le groupe *Highlight* privilégie les méthodes génériques post-hoc. On peut maintenant identifier les approches produisant les explications les plus fidèles selon chaque groupe de métriques, à savoir BR-NPA pour le groupe *Mask* et Score-CAM pour le groupe *Highlight*. Nous procédons ensuite à une évaluation qualitative des cartes de saillance produites par ces deux approches.

Étude des cartes de saillance. Nous proposons maintenant d'examiner les cartes de saillance pour comparer la pertinence biologique des indices utilisé par BR-NPA et par le CNN, expliqué

Mask			Highlight		
Average rank	Metric	Type	Average rank	Metric	Type
1.67	BR-NPA	Attention	2.75	Score-CAM	Post-Hoc
2.33	InterByParts	Attention	3.0	Ablation-CAM	Post-Hoc
4.33	B-CNN	Attention	3.25	Grad-CAM++	Post-Hoc
4.33	Ablation-CAM	Post-Hoc	3.75	RISE	Post-Hoc
5.67	Grad-CAM++	Post-Hoc	4.75	AM	Post-Hoc
5.67	Score-CAM	Post-Hoc	5.5	ABN	Attention
6.67	ABN	Attention	5.5	B-CNN	Attention
6.67	RISE	Post-Hoc	8.0	BR-NPA	Attention
7.67	AM	Post-Hoc	8.5	InterByParts	Attention

TABLE 6.2 – Classement moyen des méthodes selon les deux groupes de métriques : *Mask* et *Highlight*. Les modèles d'attention et les méthodes post-hoc sont respectivement mis en évidence en orange et en cyan.

par Score-CAM. Les figures 6.4 à 6.6 montrent des exemples de cartes de saillance générées par BR-NPA et Score-CAM au cours de 3 phases de développement de l'embryon : pPNa (figure 6.4), p4 (figure 6.5) et pB (figure 6.6). Globalement, les cartes de BR-NPA sont plus focalisées que celles de Score-CAM, ce qui est confirmée par les valeurs moyennes de parcimonie indiquées dans le tableau 6.3. Dans le cas de la figure 6.4, BR-NPA se concentre distinctement sur les Pro-Nuclei (PN) (les noyaux d'un spermatozoïde ou d'un ovule pendant le processus de fertilisation) pendant le stade pPNa alors que le CNN semble se concentrer à un niveau supérieur puisqu'il met en évidence l'ensemble de l'embryon. La focalisation de l'attention de BR-NPA sur les PN est biologiquement pertinente car la visibilité des PN est une caractéristique principale de ce stade. En figure 6.6, on peut voir que BR-NPA se concentre sur la paroi de l'embryon pendant la phase pB. Ce stade marque le début d'une nouvelle structure cellulaire dans l'embryon où, au lieu d'être disposées en paquet, les cellules commencent à se spécialiser et certaines cellules forment notamment la paroi de l'embryon. Par conséquent, le fait de se concentrer sur la paroi de l'embryon est également pertinent d'un point de vue biologique car il s'agit d'un marqueur de cette phase. En revanche, Score-CAM produit des cartes à faible résolution qui ne permettent pas de mettre en évidence les détails et se concentre sur l'embryon dans son ensemble, un niveau moins pertinent sur le plan biologique étant donné qu'il est visible à tous les stades. Pour compléter l'analyse qualitative, les figures 6.7 et 6.8 montrent des exemples de cartes de saillance pour chacun des 16 stades de développement. Globalement, les cartes de saillance BR-NPA sont détaillées et mettent en évidence les éléments biologiquement pertinents de l'image. Cependant,

cela n'implique pas que le BR-NPA se concentre sur des éléments plus pertinents sur le plan biologique que le CNN mais qu'il est difficile de comprendre quels sont les éléments que le CNN utilise à cause de sa faible résolution. Le flou des cartes de saillance de Score-CAM est probablement dû au moins en partie à la faible résolution des cartes de caractéristiques du CNN (14×14) par rapport à la résolution de BR-NPA qui est plus élevée (56×56). Cela montre l'intérêt d'utiliser des cartes d'attention à haute résolution comme les cartes de BR-NPA pour réduire l'ambiguïté et faciliter l'interprétation de la décision du modèle.

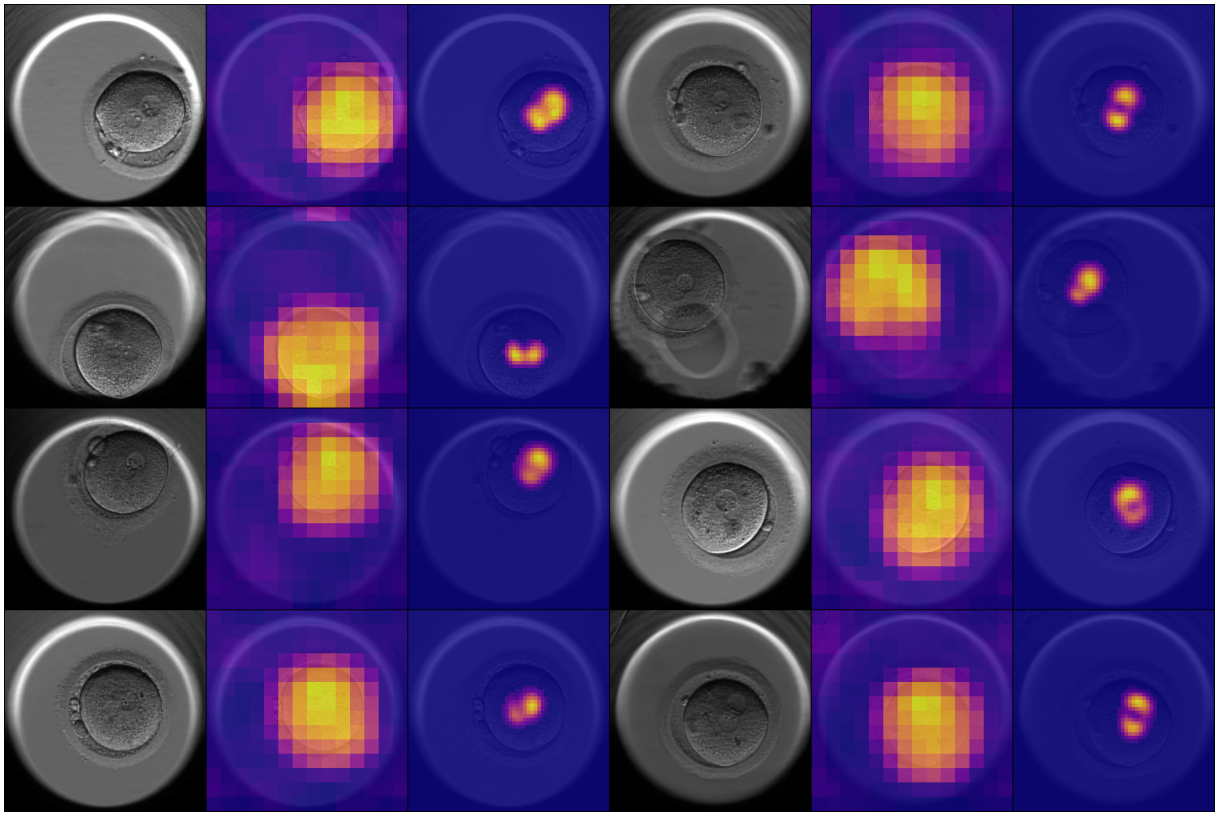


FIGURE 6.4 – Les cartes de saillance générées par la méthode Score-CAM et le modèle BR-NPA (respectivement colonnes du milieu et de droite) pendant l'étape pPNa.

6.5 Discussion

Bastings et al. [10] argumentent que les méthodes post-hoc devraient être privilégiées par rapport aux modèles d'attention lorsqu'il s'agit de fidélité, car les méthodes post-hoc prennent en compte l'ensemble de l'inférence alors que les cartes d'attention ne reflètent l'importance de

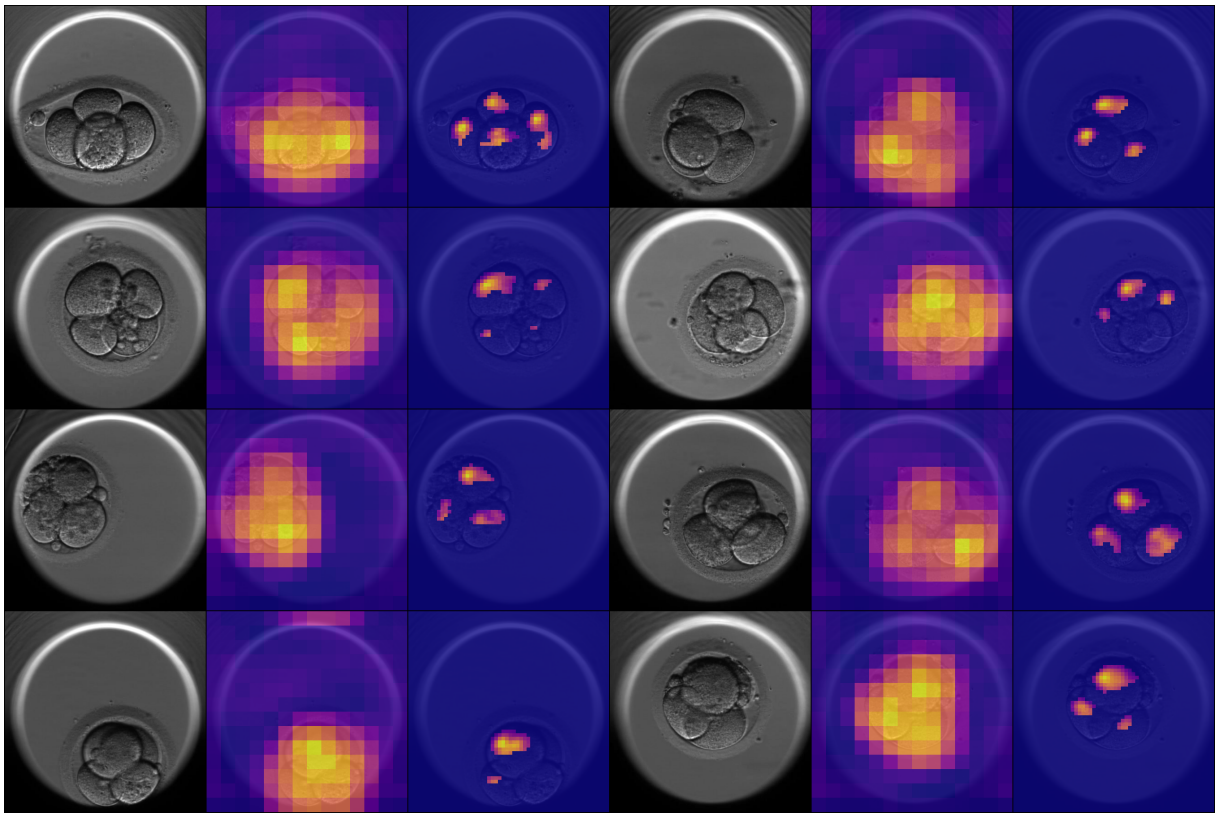


FIGURE 6.5 – Cartes de saillance générées par la méthode Score-CAM et le modèle BR-NPA (respectivement colonnes du milieu et de droite) pendant l'étape p4.

Modèle	Méthode de vis.	Stade de développement		
		pPNa	p4	pB
CNN	Score-CAM	2.94	2.91	2.56
BR-NPA	-	4.67	4.69	4.09

TABLE 6.3 – Valeurs de parcimonie moyennes sur les images de les figures 6.4 à 6.6. Sur les trois stades de développement étudiés, les cartes de BR-NPA sont celles avec la plus haute parcimonie.

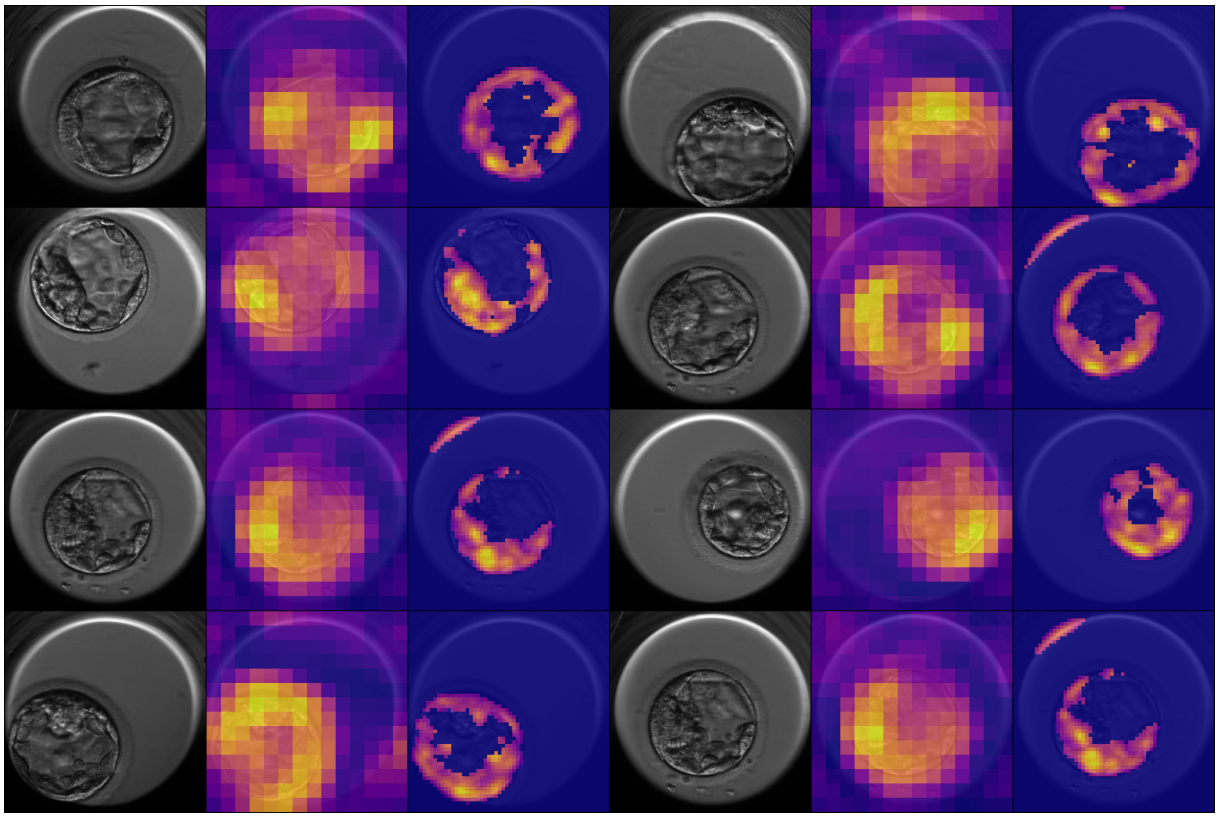


FIGURE 6.6 – Cartes de saillance générées par la méthode Score-CAM et le modèle BR-NPA (respectivement colonnes du milieu et de droite) pendant l'étape pB.

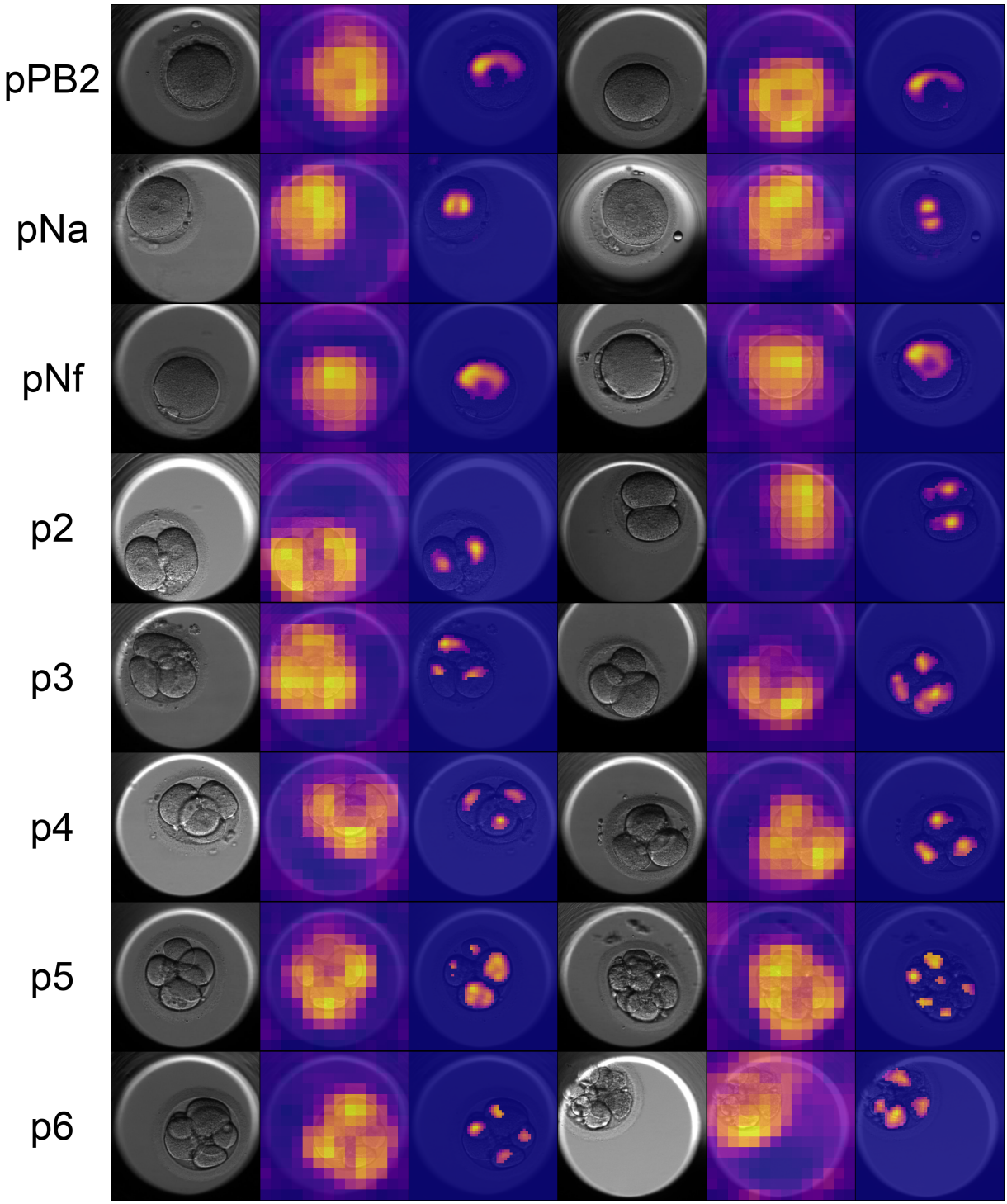


FIGURE 6.7 – Cartes de saillance générées pour les étapes pPB2 à p6

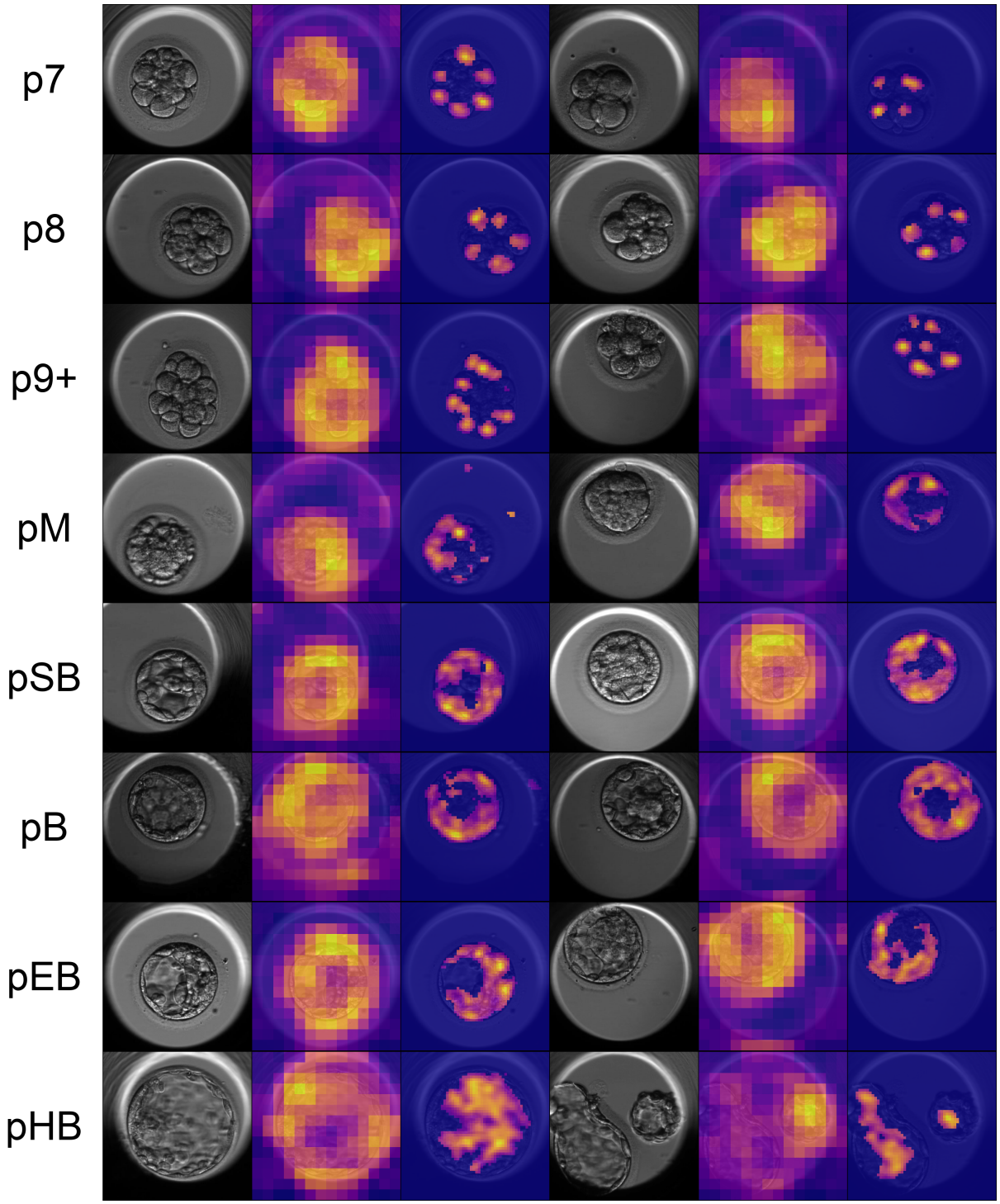


FIGURE 6.8 – Cartes de saillance générées pour les étapes p7 à pHB

l'entrée qu'à un moment de l'inférence. Cependant, nous avons montré ici que selon le type de métrique utilisée, les modèles d'attention peuvent fournir une fidélité supérieure.

Par ailleurs, les intuitions derrière les métriques discutées ici devraient être examinées. Par exemple, la métrique IIC suggère que le score de classe devrait augmenter une fois que les parties saillantes de l'image ont été mises en évidence. Cependant, il n'est pas clair si un tel phénomène se produit dans la pratique. En effet, le remplissage des zones masquées avec des pixels noirs génère des échantillons hors distribution qui peuvent induire un comportement inattendu du modèle [48]. De plus, l'ICC suggère que la confiance est augmentée par la présence de zones saillantes et diminuée par la présence de zones non saillantes. Or, les zones non saillantes pourraient aussi jouer un rôle neutre et ne pas affecter la confiance, auquel cas les masquer n'affecterait pas la confiance.

Plus généralement, la difficulté majeure dans la conception de ces métriques réside dans la mesure fiable de l'impact d'une zone de l'image d'entrée. Mesurer l'impact d'une zone en la masquant est difficile car la variation du score dépend de quelles autres zones de l'image sont masquées ou non. Pour parvenir à un consensus sur la manière dont cette mesure devrait être effectuée, la communauté doit d'abord comprendre comment la variation du score dépend exactement du masquage des autres zones. De plus, les métriques étudiées ici ont toutes été développées pour quantifier la fidélité mais en pratique, elles adoptent deux comportements distincts et semblent favoriser des approches différentes : les modèles d'attention pour le groupe *Mask* et les méthodes post-hoc pour le groupe *Highlight*. Par conséquent, des travaux futurs devraient également porter sur la définition de la fidélité et sa relation avec l'approche (modèles d'attention ou méthodes post-hoc).

Enfin, la pertinence pratique des mesures de fidélité devrait être évaluée dans le cadre d'une étude avec des utilisateurs. Dans quelle mesure la fidélité des cartes peut améliorer, par exemple, l'acceptabilité de la décision par un utilisateur reste une question ouverte.

6.6 Conclusion

Dans ce chapitre, nous avons comparé la fidélité des cartes de saillance générées par 9 approches différentes, dont 4 modèles d'attention et 5 méthodes d'explication post-hoc. Nous avons montré une faible concordance globale entre les métriques proposées dans la littérature et avons notamment démontré la tendance des métriques du groupe *Highlight* à favoriser les méthodes post-hoc alors que les métriques du groupe *Mask* favorisent plutôt les modèles d'attention. Nous avons ensuite visualisé les cartes de saillance générées par les deux meilleures

approches, à savoir Score-CAM et BR-NPA, et nous avons montré que la faible résolution de Score-CAM limite la compréhension que l'on peut en avoir, alors que les cartes de BR-NPA mettent en évidence les caractéristiques biologiquement pertinentes. Enfin, nous avons discuté de la difficulté de mesurer de manière fiable l'impact de chaque partie de l'image et avons énuméré divers aspects qui restent à travailler, comme la définition de la fidélité et sa relation avec l'approche (modèles d'attention ou méthodes post-hoc). Ces travaux ont fait l'objet d'une publication dans une conférence internationale avec comité de lecture [47].

À retenir

— Le type de métrique (*Highlight* ou *Mask*) détermine le type d'approche favorisé (méthodes post-hoc ou modèle d'attention).
— BR-NPA et Score-CAM sont les approches les plus fiables parmi celles étudiées.
— La haute résolution des cartes de BR-NPA permet de comprendre quels sont les indices visuels utilisés.

CONCLUSION DE LA SECONDE PARTIE

Nous avons d'abord vu en chapitre 4 que les précédentes applications d'apprentissage profond à la FIV ont deux limites. Premièrement, les données utilisées sont privées et les modèles ne sont entraînés à modéliser qu'un nombre réduit de phases. Deuxièmement, seulement une étude a déjà proposé d'utiliser une architecture interprétable pour les images d'embryons time-lapse.

Nous avons donc proposé dans le chapitre 5 une base de référence constituée d'un grand nombre de vidéos time-lapse d'embryons complètes accompagnées d'annotations détaillées. Les annotations recouvrent tous les stades de développement de l'embryon du premier au cinquième jour et permettent d'enrichir les prédictions faites par le modèle en comparaison aux travaux précédents. Nous avons observé le gain de performance apporté par les modèles d'AP conçus pour le format vidéo par rapport à l'approche *ad-hoc* précédente, confirmant l'intérêt d'un jeu de données constituées de vidéos complètes suffisamment grand pour l'AP.

Ensuite, nous avons appliqué en chapitre 6 le modèle BR-NPA sur cette base et avons utilisé les métriques de fiabilité pour comparer l'interprétabilité de BR-NPA à celles d'autres modèles interprétables et méthodes d'explications post-hoc de la littérature. Nous avons observé que les métriques de fiabilité de type *Highlight* favorise les méthodes d'explication post-hoc alors que les métriques de type *Mask* favorisent les modèles d'attention. Ensuite, nous avons montré que BR-NPA et l'algorithme Score-CAM sont les meilleures approches respectivement selon les métriques de type *Mask* et *Highlight*. Enfin, nous avons montré que la haute résolution des cartes de saillance permet de réduire l'ambiguïté d'une carte en permettant au modèle de se concentrer sur des zones plus réduites. Une étude qualitative montre que BR-NPA concentre son attention sur des indices visuels de taille réduits pertinents pour l'identification de ces stades de développement. Au contraire, les cartes de Score-CAM recouvrent une grande partie de l'image et ne permettent pas d'interpréter clairement comment se distribue l'attention.

CHAPITRE 7

CONCLUSION ET PERSPECTIVES

7.1 Conclusion

Dans la première partie de cette thèse, nous avons présenté des méthodes et modèles pour obtenir des explications visuelles sous la forme de carte de saillance ainsi que des méthodologies d'évaluation de ces cartes. D'abord, nous avons étudié les modèles d'attention et les méthodes d'explication de la littérature permettant de générer des cartes de saillance. Nous avons aussi évoqué les différentes méthodes d'évaluation de la qualité de ces explications, et nous nous sommes particulièrement focalisés sur les métriques de fiabilité. Ensuite, nous avons mis en valeurs certaines limites de ces métriques et conclu notamment à une incompatibilité entre la conception des modèles de classification, qui partent du principe qu'il y a forcément un objet à classer dans l'image, et la conception des métriques, qui consistent à effacer cet objet. Enfin, nous avons présenté un nouveau modèle d'attention non paramétrique générant des cartes hautes résolution appelé BR-NPA. À travers des évaluations qualitatives et quantitatives, nous avons montré que cette architecture produit des cartes fiables à bas coût sans sacrifier la performance de classification qui permettent de voir précisément quels sont les indices visuels utilisés par le modèle.

Dans la deuxième partie, nous avons appliqué ces modèles, méthodes et métriques aux vidéos time-lapse issues de systèmes TLI utilisés en FIV. D'abord, nous avons montré que les progrès en apprentissage machine appliqué à la FIV sont actuellement limités par deux facteurs. Premièrement, il n'y a pas de base de référence publique permettant à la communauté de comparer leurs propositions et d'arriver à un consensus. Deuxièmement, il n'y a eu que très peu de travaux proposant de générer des cartes de saillance pour expliquer des modèles de vidéos time-lapse. Face à ces constats, nous avons d'abord proposé une base de référence pour des modèles de prédiction des stades de développement embryonnaire. Ce jeu de données est assez grand pour entraîner des modèles profonds et permet d'exploiter le contexte temporel et spatial grâce à l'inclusion des vidéos au complet et de tous les plans focaux. Nous avons ensuite procédé à l'évaluation des cartes de saillance de plusieurs méthodes post-hoc et modèles d'attention

sur cette base à l'aide de 7 métriques de fiabilité différentes. Cette expérience a montré que le type de la métrique influence largement le type d'approche favorisé par cette dernière. Après avoir identifié BR-NPA et Score-CAM comme les approches les plus fiables par les familles de métriques *Mask* et *Highlight* respectivement, nous avons effectué une évaluation qualitative des cartes de saillance produites par ces deux approches. Cette analyse a mis en valeur l'intérêt de BR-NPA et de la haute-résolution en général pour réduire l'ambiguïté des cartes de saillance et permettre d'identifier les éléments pris en compte par le modèle.

Nous avons donc proposé des contributions à trois niveaux : au niveau de l'évaluation, au niveau des modèles et au niveau des données. La combinaison des nouvelles métriques, d'un nouveau mécanisme d'attention et d'une nouvelle base de référence nous a permis de comparer pour la première fois la fiabilité des explications produites par diverses approches à partir de modèle de vidéos time-lapse.

Imaginons qu'au lieu d'utiliser des patchs noirs ou flous on utilise des patchs d'une autre image d'une autre classe.

Dans cette thèse nous avons réalisé la première étude d'interprétabilité appliquée à des données issues de systèmes TLI. Ce travail constitue la première étape pour pouvoir mettre en place un jour des modèles interprétables dans un laboratoire de FIV mais peut aussi trouver des applications dans grand nombre de situations d'imagerie médicale où il y a un besoin en interprétabilité.

7.2 Perspectives

Perspectives algorithmiques. La première perspective se situe dans le format des données utilisées. Les embryons sont des structures cellulaires en trois dimensions mais dans ce manuscrit, nous nous sommes restreints à des analyses en deux dimensions, i.e. à la classification d'image. Tenir compte de cette structure tri-dimensionnelles permettrait probablement d'améliorer les performances des modèles mais aussi l'interprétabilité de la décision. Entraîner un modèle à partir de la structure 3D de l'embryon (sous la forme d'un graphe par exemple) au lieu de l'image brute permettrait d'identifier plus clairement quelles parties de l'embryon sont exploitées pour la prédiction. Cependant, comme tous les problèmes "2D vers 3D", inférer la structure de l'embryon est une tâche difficile qui peut demander des annotations détaillées coûteuses à obtenir [59]. Une solution serait donc d'utiliser des méthodes d'apprentissage semi-supervisées ou non-supervisées comme cela est fait dans le cadre d'autres problèmes de type "2D vers 3D" [104, 90].

Deuxièmement, nous ne nous sommes pas intéressés au problème de la prédiction de l'issue

de la FIV. Ce problème consiste à prédire si la procédure de FIV sera un succès ou non à partir de la vidéo time-lapse de l'embryon. Nous n'avons pas traité ce problème dans cette thèse pour plusieurs raisons. Tout d'abord pour des raisons techniques : cette tâche requiert une grande quantité de données auxquelles nous n'avons eu accès que très tard durant cette thèse. Ensuite, cette tâche est difficile car l'issue de la FIV dépend de la qualité embryonnaire mais aussi du milieu utérin dans lequel l'embryon devra s'implanter. Ce second facteur a une importance considérable et n'est pas accessible au modèle qui traite simplement la vidéo time-lapse. Par conséquence, les modèles entraîné à cette tâche ont de faibles performances, avec une aire sous la courbe ROC proche de 0.7 [12]. Étant donné qu'une partie importante des indices n'est pas accessible dans la vidéo, il serait alors difficile d'évaluer à quel point les indices utilisés par le modèle sont pertinents, comme nous l'avons fait pour des tâches simples dans les chapitres 3 et 6. En effet, les architectures interprétables évoquées dans ce manuscrit et de façon plus générale toutes les architectures basées sur des DNN, font l'hypothèse qu'il existe des éléments pertinents dans l'image pour résoudre la tâche. Dans le chapitre 2 nous avons suggéré qu'une des causes de la nature OOD des images passées au modèle lors du calcul des métriques vient de cette même hypothèse, car les patchs appliqués sur l'image peuvent masquer totalement l'objet d'intérêt. Un modèle conçu sans cette hypothèse pourrait donc indiquer que l'image ne contient pas d'indice pertinent pour la tâche et pourrait servir à déterminer par exemple si l'échec de la FIV provient de l'embryon ou d'autres facteurs externes.

Perspectives d'évaluation. Les métriques de fiabilité permettent d'identifier les approches générant les cartes de saillance les plus fidèles à la décision prise par le modèle. Elles permettent donc d'identifier les meilleures approches au niveau de la première étape de communication entre un modèle et un utilisateur, comme illustré en figure 7.1. Cependant, nous avons vu que les métriques présentent des comportements distincts malgré le fait qu'elles sont conçues pour le même objectif. On peut supposer que ce problème est lié à un autre, celui des images OOD. Imaginons qu'au lieu d'utiliser des patchs noirs ou flous on utilise des patchs d'une autre image d'une autre classe. Il est possible que cela permette de résoudre au moins partiellement le problème d'image OOD mais aussi que cela permette d'uniformiser le comportement des métriques. En modifiant DAUC et IAUC de cette façon, la séquence d'images passée à un modèle pour le calcul de DAUC correspondrait à passer progressivement d'une image A à une image B en commençant par les zones saillantes de A et la séquence d'images de IAUC correspondrait à passer de l'image B à l'image A en commençant par les mêmes zones. On peut supposer ainsi qu'en utilisant des séquences d'images similaires, ces deux métriques aient des comportements

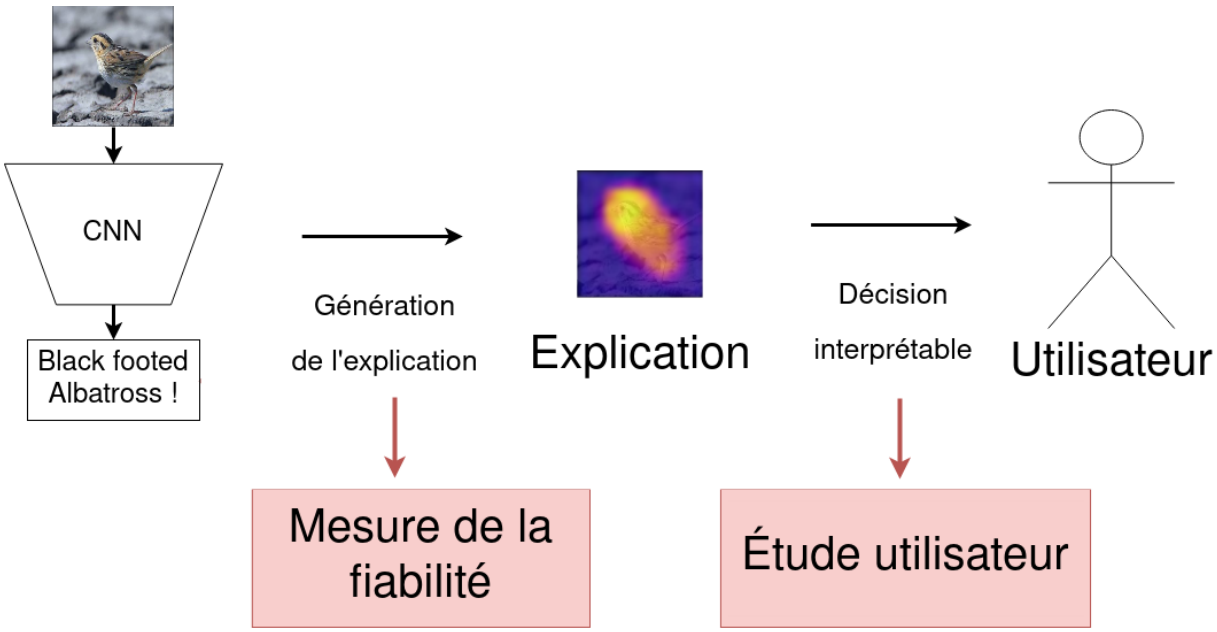


FIGURE 7.1 – La communication entre un modèle et un utilisateur en deux étapes. Les métriques de fiabilité permettent d'évaluer la qualité de la communication entre le modèle et l'explication et les études utilisateur permettent d'étudier la communication entre l'explication et l'utilisateur, c-à-d. la perception de l'explication par les utilisateurs.

plus proches.

Perspectives scientifiques. Dans cette thèse, nous avons notamment cherché à évaluer la qualité de la première étape de communication entre l'utilisateur et le modèle, à savoir l'étape qui lie le modèle et l'explication (cf. figure 7.1). Pour garantir une communication efficace entre un utilisateur et un modèle, il faut aussi étudier la seconde étape, celle qui lie l'explication à l'utilisateur. Comprendre cette seconde étape demande à étudier la perception des explications par les utilisateurs. Plus précisément, il faut identifier les facteurs qui jouent un rôle important dans la perception du modèle par l'utilisateur. On peut supposer que ces facteurs soient déterminés par le modèle mais aussi par la nature de l'explication. Pour étudier ces facteurs, plusieurs auteurs dans la littérature ont proposé d'évaluer la capacité de l'utilisateur à réaliser certaines tâches en fonction de la complexité de l'explication fournie notamment [106, 86]. Par exemple, Narayanan et al. [106] demandent aux utilisateurs de vérifier la cohérence entre la prédiction d'un modèle de recommandation et une explication sous la forme d'ensemble de règles. On peut aussi citer Lage et al. [86] qui proposent une expérience similaire et demandent à l'utilisateur de répondre à des

questions contrefactuelles [1] ou d'exécuter d'autres tâches comme la simulation (prédire la sortie du modèle). Ces deux travaux montrent notamment que le taux d'erreur de l'utilisateur semble être peu affecté par la complexité de l'explication et que celle-ci est corrélée positivement et négativement respectivement avec le temps de réponse et la satisfaction de l'utilisateur.

Des questions similaires pourraient alors être posées dans le domaine de la classification d'image expliquées par des cartes de saillance. Par exemple, nous pouvons supposer que dans le cas d'un modèle produisant plusieurs cartes d'attention par image, le fait de montrer toutes les cartes à l'utilisateur augmente le temps de réponse sans toutefois lui permettre de réduire significativement son taux d'erreur. Mais il existe aussi probablement des facteurs d'influence propres à la nature visuelle des cartes de saillance. Par exemple, il est possible qu'une explication parcimonieuse soit perçue comme une preuve de la précision du modèle et qu'au contraire une explication peu parcimonieuse indique aux yeux de l'utilisateur que le modèle ne produit pas des prédictions fiables.

Il est aussi possible que la manière avec laquelle est présentée une carte de saillance à l'utilisateur joue un rôle important. Dans cette thèse, nous avons représenté les cartes de saillance en les superposant à l'image d'entrée et en utilisant la fonction de couleur "plasma" de la bibliothèque Matplotlib [70] (ou simplement en faisant varier l'intensité des couleurs pour représenter chaque carte dans le cadre des modèles d'attention multi-cartes). Il est probable que ces choix affectent la perception de l'utilisateur et l'on pourrait donc identifier ici un autre exemple de facteur d'influence propre à l'utilisation de cartes de saillance. Le fait de superposer l'image d'entrée à la carte de saillance peut permettre de mieux identifier les points d'intérêt du modèle mais peut aussi en cacher d'autres si une partie de l'attention se focalise sur des endroits sombres de l'image d'entrée. La fonction de couleur utilisée a probablement aussi un rôle important dans la mesure ou toutes les couleurs ne sont pas perçues de la même manière par l'utilisateur [95].

Nous nous situons dans le cadre de tâches visuelles nécessitant une expertise, et l'on pourrait donc exploiter l'expertise humaine pour entraîner des modèles à se focaliser sur des indices visuels pertinents. Plus précisément, on pourrait demander à un expert d'annoter manuellement les zones pertinentes de l'image et ensuite apprendre à un modèle à imiter ces cartes de saillance "vérité-terrain" grâce à un apprentissage supervisé (contrairement aux modèles entraînés durant cette thèse, où l'attention est apprise de façon semi-supervisée). Une telle procédure est cependant peu envisageable car elle demanderait aux experts d'y investir beaucoup de temps. Une alternative serait de construire des cartes à partir de leur attention visuelle à l'aide d'un oculomètre. Cela

1. Des questions où l'utilisateur doit deviner si la sortie changerait après une modification donnée de l'entrée

permettrait de ne demander aux experts que d'exécuter la tâche et réduirait le coût en temps de la procédure. Ces deux possibilités ne sont cependant valides que si une carte de saillance construite de l'une des deux manières constitue une explication utile pour un utilisateur novice [146]. De plus, la manière dont on obtiendrait ces cartes "vérité-terrain" pourrait constituer un autre facteur d'influence sur la perception de l'utilisateur.

Il faut aussi envisager que les cartes de saillance ne soient pas suffisantes pour permettre à l'utilisateur de comprendre un modèle, possiblement parce qu'une carte ne fait qu'indiquer les zones importantes et laisse à l'utilisateur le soin de les rattacher à une classe particulière. Par exemple, Alqaraawi et al. [7] ont montré qu'il est difficile pour un utilisateur de prédire la sortie d'un DNN de classification d'image même à l'aide d'une carte de saillance et argumentent pour étendre la recherche à d'autres formes d'explications. L'utilisation de prototypes semble donc être une solution potentielle à ce problème en aidant l'utilisateur à associer les zones saillantes de l'image à une classe. Il faudrait donc étudier l'impact des prototypes combinés aux cartes de saillance sur la compréhension du modèle par l'utilisateur par rapport à une utilisation des cartes seules.

Pour répondre à ces questions, on pourra s'inspirer des travaux de Narayanan et al. [106] et Lage et al. [86] pour construire un protocole pertinent tout en prenant en compte le fait que les cartes de saillance ne constituent peut-être pas des explications suffisantes utilisées seules [7]. L'étude de ces questions et de celles posées par cette thèse est primordiale pour pouvoir donner tort à Alan Turing [2] et garantir un jour une communication efficace entre modèle et utilisateur.

2. Voir citation au début de ce manuscrit.

BIBLIOGRAPHIE

[1] Mehryar ABBASI et al., « A Deep Learning Approach for Prediction of IVF Implantation Outcome from Day 3 and Day 5 Time-Lapse Human Embryo Image Sequences », in : *2021 IEEE International Conference on Image Processing (ICIP)*, 2021, p. 289-293, DOI : 10.1109/ICIP42928.2021.9506097.

[2] Julius ADEBAYO et al., *Local Explanation Methods for Deep Neural Networks Lack Sensitivity to Parameter Values*, 2018, arXiv : 1810.03307 [cs.CV].

[3] Julius ADEBAYO et al., « Sanity Checks for Saliency Maps », in : *Advances in Neural Information Processing Systems*, sous la dir. de S. BENGIO et al., t. 31, Curran Associates, Inc., 2018.

[4] M. AFNAN et et AL., « Interpretable, not black-box, artificial intelligence should be used for embryo selection », in : *Human Reproduction Open* 2021.*4* (nov. 2021), hoab040, DOI : 10.1093/hropen/hoab040, eprint : https://academic.oup.com/hropen/article-pdf/2021/4/hoab040/41285975/hoab040.pdf.

[5] Takuya AKIBA et al., *Optuna A Next-generation Hyperparameter Optimization Framework*, 2019, arXiv : 1907.10902 [cs.LG].

[6] Ahmed ALQARAAWI et al., « Evaluating Saliency Map Explanations for Convolutional Neural Networks : A User Study », in : IUI '20, Cagliari, Italy : Association for Computing Machinery, 2020, p. 275-285, ISBN : 9781450371186, DOI : 10.1145/3377325.3377519.

[7] Ahmed ALQARAAWI et al., « Evaluating Saliency Map Explanations for Convolutional Neural Networks : A User Study », in : *Proceedings of the 25th International Conference on Intelligent User Interfaces*, IUI '20, Cagliari, Italy : Association for Computing Machinery, 2020, p. 275-285, ISBN : 9781450371186, DOI : 10.1145/3377325.3377519, URL : https://doi.org/10.1145/3377325.3377519.

[8] S ARMSTRONG et al., « Time-lapse systems for embryo incubation and assessment in assisted reproduction », in : *Cochrane Database of Systematic Reviews 5* (2019), ISSN : 1465-1858, DOI : 10.1002/14651858.CD011320.pub4.

[9] Sebastian BACH et al., « On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation », in : *PLOS ONE* 10.*7* (juill. 2015), p. 1-46, DOI : 10.1371/journal.pone.0130140, URL : https://doi.org/10.1371/journal.pone.0130140.

[10] Jasmijn BASTINGS et Katja FILIPPOVA, « The elephant in the interpretability room : Why use attention as explanation when we have saliency methods ? », in : *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, Online : Association for Computational Linguistics, nov. 2020, p. 149-155, DOI : 10.18653/v1/2020.blackboxnlp-1.14.

[11] A. E. BAXTER BENDUS et al., « Interobserver and intraobserver variation in day 3 embryo grading », in : *Fertil Steril* 86.*6* (déc. 2006), p. 1608-1615.

[12] Jørgen BERNTSEN et al., *Robust and generalizable embryo selection based on artificial intelligence and time-lapse image sequences*, 2021, arXiv : 2103.07262 [cs.LG].

[13] Jørgen BERNTSEN et al., « Robust and generalizable embryo selection based on artificial intelligence and time-lapse image sequences », in : *PLOS ONE* 17.*2* (fév. 2022), p. 1-18, DOI : 10.1371/journal.pone.0262661, URL : https://doi.org/10.1371/journal.pone.0262661.

[14] Luca BERTINETTO et al., « Meta-learning with differentiable closed-form solvers », in : *International Conference on Learning Representations*, 2019.

[15] Tom B. BROWN et al., *Language Models are Few-Shot Learners*, 2020, arXiv : 2005.14165 [cs.CL].

[16] Oana-Maria CAMBURU et al., « The Struggles of Feature-Based Explanations : Shapley Values vs. Minimal Sufficient Subsets », in : (2020), DOI : 10.48550/ARXIV.2009.11023, URL : https://arxiv.org/abs/2009.11023.

[17] A CAMPBELL et al., « O-125 Application of artificial intelligence using big data to devise and train a machine learning model on over 63,000 human embryos to automate time-lapse embryo annotation », in : *Human Reproduction* 37.*Supplement_1* (juin 2022), deac105.025, ISSN : 0268-1161, DOI : 10.1093/humrep/deac105.025, eprint : https://academic.oup.com/humrep/article-pdf/37/Supplement_1/deac105.025/44306447/deac105.025.pdf, URL : https://doi.org/10.1093/humrep/deac105.025.

[18] Mathilde CARON et al., *Emerging Properties in Self-Supervised Vision Transformers*, 2021, DOI : 10.48550/ARXIV.2104.14294.

[19] A. CHATTOPADHAY et al., « Grad-CAM++ Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks », in : *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, p. 839-847, DOI : 10.1109/WACV.2018.00097.

[20] Alejandro CHAVEZ-BADIOLA et al., « Predicting pregnancy test results after embryo transfer by image feature extraction and analysis using machine learning », in : *Scientific Reports* 10.*1* (mars 2020), p. 4394, ISSN : 2045-2322, DOI : 10.1038/s41598-020-61357-9.

[21] Chaofan CHEN et al., « This looks like that : deep learning for interpretable image recognition », in : *NeurIPS*, 2019.

[22] Jianpin CHEN et al., « Attention-based cropping and erasing learning with coarse-to-fine refinement for fine-grained visual classification », in : *Neurocomputing* 501 (2022), p. 359-369, ISSN : 0925-2312, DOI : https://doi.org/10.1016/j.neucom.2022.06.041, URL : https://www.sciencedirect.com/science/article/pii/S0925231222007603.

[23] Junsuk CHOE, Seungho LEE et Hyunjung SHIM, « Attention-Based Dropout Layer for Weakly Supervised Single Object Localization and Semantic Segmentation », in : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.*12* (2021), p. 4256-4271, DOI : 10.1109/TPAMI.2020.2999099.

[24] Dahim CHOI et al., « Multidimensional noise reduction in C-arm cone-beam CT via 2D-based Landweber iteration and 3D-based deep neural networks », in : *Medical Imaging 2019 : Physics of Medical Imaging*, t. 10948, International Society for Optics et Photonics, 2019, p. 1094837.

[25] H. Nadir CIRAY et al., « Proposed guidelines on the nomenclature and annotation of dynamic human embryo monitoring by a time-lapse user group », in : *Human Reproduction* 29.*12* (oct. 2014), p. 2650-2660, ISSN : 0268-1161, DOI : 10.1093/humrep/deu278.

[26] R. COLLOBERT, K. KAVUKCUOGLU et C. FARABET, « Torch7 : A Matlab-like Environment for Machine Learning », in : *BigLearn, NIPS Workshop*, 2011.

[27] Joe CONAGHAN et al., « Improving embryo selection using a computer-automated time-lapse image analysis test plus day 3 morphology : results from a prospective multicenter trial », in : *Fertility and Sterility* 100.*2* (août 2013), 412-419.e5, ISSN : 0015-0282, DOI : 10.1016/j.fertnstert.2013.04.021.

[28] Yimian DAI et al., « Attentional Feature Fusion », in : *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021, p. 3559-3568, DOI : 10.1109/WACV48630.2021.00360.

[29] Jean-Stanislas DENAIN et Jacob STEINHARDT, *Auditing Visualizations : Transparency Methods Struggle to Detect Anomalous Behavior*, 2022, DOI : 10.48550/ARXIV.2206.13498, URL : https://arxiv.org/abs/2206.13498.

[30] Saurabh DESAI et Harish G. RAMASWAMY, « Ablation-CAM : Visual Explanations for Deep Convolutional Network via Gradient-free Localization », in : *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, p. 972-980, DOI : 10.1109/WACV45572.2020.9093360.

[31] Saurabh DESAI et Harish G. RAMASWAMY, « Ablation-CAM : Visual Explanations for Deep Convolutional Network via Gradient-free Localization », in : *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, p. 972-980, DOI : 10.1109/WACV45572.2020.9093360.

[32] Jon DONNELLY, Alina Jade BARNETT et Chaofan CHEN, « Deformable ProtoPNet : An Interpretable Image Classifier Using Deformable Prototypes », in : *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, juin 2022, p. 10265-10275.

[33] Alexey DOSOVITSKIY et al., *An Image is Worth 16x16 Words : Transformers for Image Recognition at Scale*, 2020, DOI : 10.48550/ARXIV.2010.11929.

[34] Abhimanyu DUBEY et al., *Pairwise Confusion for Fine-Grained Visual Classification*, 2018, arXiv : 1705.08016 [cs.CV].

[35] S. DYER et al., « International Committee for Monitoring Assisted Reproductive Technologies world report : Assisted Reproductive Technology 2008, 2009 and 2010† », in : *Human Reproduction* 31.*7* (mai 2016), p. 1588-1609, ISSN : 0268-1161, DOI : 10.1093/humrep/dew082, eprint : https://academic.oup.com/humrep/article-pdf/31/7/1588/17372872/dew082.pdf.

[36] Bradley EFRON et al., « Least Angle Regression », in : *The Annals of Statistics* 32.*2* (2004), p. 407-451, ISSN : 00905364, URL : http://www.jstor.org/stable/3448465 (visité le 20/06/2022).

[37] *Explicabilité*, https://www.cnil.fr/fr/definition/explicabilite, Accessed : 2022-06-20.

[38] A. P. FERRARETTI et al., « Assisted reproductive technology in Europe, 2009 : results generated from European registers by ESHRE », eng, in : *Human Reproduction (Oxford, England)* 28.*9* (sept. 2013), p. 2318-2331, ISSN : 1460-2350, DOI : 10.1093/humrep/det278.

[39] M FEYEUX et al., « Development of automated annotation software for human embryo morphokinetics », in : *Human Reproduction* 35.*3* (mars 2020), p. 557-564, DOI : 10.1093/humrep/deaa001, eprint : https://academic.oup.com/humrep/article-pdf/35/3/557/32980919/deaa001.pdf.

[40] J FJELDSTAD et al., « O-204 Non-invasive AI image analysis unlocks the secrets of oocyte quality and reproductive potential by assigning 'Magenta' scores from 2-dimensional (2-D) microscope images », in : *Human Reproduction* 37.*Supplement_1* (juin 2022), deac104.119, ISSN : 0268-1161, DOI : 10.1093/humrep/deac104.119, eprint : https://academic.oup.com/humrep/article-pdf/37/Supplement_1/deac104.119/44306319/deac104.119.pdf, URL : https://doi.org/10.1093/humrep/deac104.119.

[41] Thomas FRÉOUR et al., « External validation of a time-lapse prediction model », in : *Fertility and Sterility* 103.*4* (avr. 2015), p. 917-922, ISSN : 0015-0282, DOI : 10.1016/j.fertnstert.2014.12.111.

[42] Ruigang FU et al., *Axiom-based Grad-CAM : Towards Accurate Visualization and Explanation of CNNs*, 2020, arXiv : 2008.02312 [cs.CV].

[43] Hiroshi FUKUI et al., « Attention Branch Network : Learning of Attention Mechanism for Visual Explanation », in : *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, p. 10697-10706, DOI : 10.1109/CVPR.2019.01096.

[44] Noritaka FUKUNAGA et al., « Development of an automated two pronuclei detection system on time-lapse embryo images using deep learning techniques », in : *Reproductive Medicine and Biology* 19.*3* (2020), p. 286-294, DOI : https://doi.org/10.

1002/rmb2.12331, eprint : https://onlinelibrary.wiley.com/doi/pdf/10.1002/rmb2.12331.

[45] W. GE, X. LIN et Y. YU, « Weakly Supervised Complementary Parts Models for Fine-Grained Image Classification From the Bottom Up », in : *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, p. 3029-3038.

[46] Sanjukta GHOSH et al., « Robustness of deep convolutional neural networks for image degradations », in : *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2018, p. 2916-2920.

[47] Tristan GOMEZ, Thomas FRÉOUR et Harold MOUCHÈRE, « Comparison of attention models and post-hoc explanation methods for embryo stage identification : a case study », in : *XAI Workshop (ICPR 2022)*, Montréal, Canada, août 2022, URL : https://hal.archives-ouvertes.fr/hal-03690574.

[48] Tristan GOMEZ, Thomas FRÉOUR et Harold MOUCHÈRE, « Metrics for Saliency Map Evaluation of Deep Learning Explanation Methods », in : *Pattern Recognition and Artificial Intelligence*, sous la dir. de Mounîm EL YACOUBI et al., Cham : Springer International Publishing, 2022, p. 84-95, ISBN : 978-3-031-09037-0.

[49] Tristan GOMEZ et al., « A time-lapse embryo dataset for morphokinetic parameter prediction », in : *Data in Brief* 42 (2022), p. 108258, ISSN : 2352-3409, DOI : https://doi.org/10.1016/j.dib.2022.108258, URL : https://www.sciencedirect.com/science/article/pii/S2352340922004607.

[50] Tristan GOMEZ et al., « BR-NPA : A non-parametric high-resolution attention model to improve the interpretability of attention », in : *Pattern Recognition* 132 (2022), p. 108927, ISSN : 0031-3203, DOI : https://doi.org/10.1016/j.patcog.2022.108927, URL : https://www.sciencedirect.com/science/article/pii/S0031320322004083.

[51] A. GRAVES, N. JAITLY et A. MOHAMED, « Hybrid speech recognition with Deep Bidirectional LSTM », in : *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, déc. 2013, p. 273-278, DOI : 10.1109/ASRU.2013.6707742.

[52] Riccardo GUIDOTTI, « Counterfactual explanations and how to find them : literature review and benchmarking », in : *Data Mining and Knowledge Discovery* (avr. 2022), ISSN : 1573-756X, DOI : 10.1007/s10618-022-00831-6, URL : https://doi.org/10.1007/s10618-022-00831-6.

[53] Chuan GUO et al., « On Calibration of Modern Neural Networks », in : *Proceedings of the 34th International Conference on Machine Learning*, sous la dir. de Doina PRECUP et Yee Whye TEH, t. 70, Proceedings of Machine Learning Research, PMLR, juin 2017, p. 1321-1330.

[54] Seungwook HAN et al., « 3D distributed deep learning framework for prediction of human intelligence from brain MRI », in : *Medical Imaging 2020 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, sous la dir. d'Andrzej KROL et Barjor S. GIMI, t. 11317, International Society for Optics et Photonics, SPIE, 2020, p. 484-490, DOI : 10.1117/12.2549758.

[55] Kensho HARA, Hirokatsu KATAOKA et Yutaka SATOH, « Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition », in : *CoRR* abs/1708.07632 (2017), arXiv : 1708.07632.

[56] M. HARANDI et al., « Dictionary Learning and Sparse Coding on Grassmann Manifolds : An Extrinsic Solution », in : *2013 IEEE International Conference on Computer Vision (ICCV)*, Los Alamitos, CA, USA : IEEE Computer Society, déc. 2013, p. 3120-3127, DOI : 10.1109/ICCV.2013.387, URL : https://doi.ieeecomputersociety.org/10.1109/ICCV.2013.387.

[57] Chunliu HE et al., « Automated classification of coronary plaque calcification in OCT pullbacks with 3D deep neural networks », in : *Journal of Biomedical Optics* 25.*9* (2020), p. 1-13, DOI : 10.1117/1.JBO.25.9.095003.

[58] K. HE et al., « Deep Residual Learning for Image Recognition », in : *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, p. 770-778.

[59] P HE et al., « O-177 Towards 3D Reconstructions of Human Preimplantation Embryo Development », in : *Human Reproduction* 37.*Supplement_1* (juin 2022), deac105.091, ISSN : 0268-1161, DOI : 10.1093/humrep/deac105.091, eprint : https://academic.oup.com/humrep/article-pdf/37/Supplement_1/deac105.091/44305755/deac105.091.pdf, URL : https://doi.org/10.1093/humrep/deac105.091.

[60] Geoffrey HINTON, Oriol VINYALS et Jeff DEAN, *Distilling the Knowledge in a Neural Network*, 2015, arXiv : 1503.02531 [stat.ML].

[61] Sepp HOCHREITER et Jürgen SCHMIDHUBER, « Long Short-Term Memory », in : *Neural Comput.* 9.*8* (nov. 1997), p. 1735-1780.

[62] Ruibing HOU et al., *Cross Attention Network for Few-shot Classification*, 2019, arXiv : 1910.07677 [cs.CV].

[63] Jie HU, Li SHEN et Gang SUN, « Squeeze-and-Excitation Networks », in : *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, p. 7132-7141, DOI : 10.1109/CVPR.2018.00745.

[64] Tao HU et Honggang QI, « See Better Before Looking Closer : Weakly Supervised Data Augmentation Network for Fine-Grained Visual Classification », in : *CoRR* abs/1901.09891 (2019), arXiv : 1901.09891.

[65] Bo HUANG et al., « Using deep learning to predict the outcome of live birth from more than 10,000 embryo data », in : *BMC Pregnancy and Childbirth* 22.*1* (jan. 2022), p. 36, ISSN : 1471-2393, DOI : 10.1186/s12884-021-04373-5, URL : https://doi.org/10.1186/s12884-021-04373-5.

[66] Thomas T.F. HUANG et al., « Deep learning neural network analysis of human blastocyst expansion from time-lapse image files », in : *Reproductive BioMedicine Online* 42.*6* (2021), p. 1075-1085, ISSN : 1472-6483, DOI : https://doi.org/10.1016/j.rbmo.2021.02.015, URL : https://www.sciencedirect.com/science/article/pii/S1472648321001012.

[67] Zixuan HUANG et Yin LI, *Interpretable and Accurate Fine-grained Recognition via Region Grouping*, 2020, arXiv : 2005.10411 [cs.CV].

[68] Zixuan HUANG et Yin LI, *Interpretable and Accurate Fine-grained Recognition via Region Grouping*, 2020, arXiv : 2005.10411 [cs.CV].

[69] Zixuan HUANG et Yin LI, « Interpretable and Accurate Fine-grained Recognition via Region Grouping », in : *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, juin 2020.

[70] J. D. HUNTER, « Matplotlib : A 2D graphics environment », in : *Computing in Science & Engineering* 9.*3* (2007), p. 90-95, DOI : 10.1109/MCSE.2007.55.

[71] Ashiq IMRAN et Vassilis ATHITSOS, « Domain Adaptive Transfer Learning on Visual Attention Aware Data Augmentation for Fine-Grained Visual Categorization », in : *Advances in Visual Computing : 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II*, San Diego, CA, USA : Springer-Verlag, 2020, p. 53-65, ISBN : 978-3-030-64558-8, DOI : 10.1007/978-3-030-64559-5_5.

[72] Marcia C. INHORN et Pasquale PATRIZIO, « Infertility around the globe : new thinking on gender, reproductive technologies and global movements in the 21st century », in : *Human Reproduction Update* 21.*4* (mars 2015), p. 411-426, ISSN : 1355-4786, DOI : 10.1093/humupd/dmv016, eprint : https://academic.oup.com/humupd/article-pdf/21/4/411/1995247/dmv016.pdf.

[73] Max JADERBERG et al., « Spatial Transformer Networks », in : *Advances in Neural Information Processing Systems 28*, sous la dir. de C. CORTES et al., Curran Associates, Inc., 2015, p. 2017-2025.

[74] Sarthak JAIN et Byron C. WALLACE, « Attention is not Explanation », in : *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota : Association for Computational Linguistics, juin 2019, p. 3543-3556, DOI : 10.18653/v1/N19-1357, URL : https://aclanthology.org/N19-1357.

[75] Debesh JHA et al., « Kvasir-seg : A segmented polyp dataset », in : *International Conference on Multimedia Modeling*, Springer, 2020, p. 451-462.

[76] Alistair E.W. JOHNSON et al., « MIMIC-III, a freely accessible critical care database », in : *Scientific Data* 3.*1* (mai 2016), p. 160035, ISSN : 2052-4463, DOI : 10.1038/sdata.2016.35.

[77] Hyungsik JUNG et Youngrock OH, « LIFT-CAM : Towards Better Explanations for Class Activation Mapping », in : *ArXiv* abs/2102.05228 (2021).

[78] Manoj Kumar KANAKASABAPATHY et al., « Development and evaluation of inexpensive automated deep learning-based imaging systems for embryology », in : *Lab Chip* 19 (24 2019), p. 4139-4145, DOI : 10.1039/C9LC00721K.

[79] M. G. KENDALL, « The Treatment of Ties in Ranking Problems », in : *Biometrika* 33.*3* (1945), p. 239-251, ISSN : 00063444, (visité le 12/04/2022).

[80] Aisha KHAN, Stephen GOULD et Mathieu SALZMANN, « Deep Convolutional Neural Networks for Human Embryonic Cell Counting », in : *Computer Vision – ECCV 2016 Workshops*, sous la dir. de Gang HUA et Hervé JÉGOU, Cham : Springer International Publishing, 2016, p. 339-348, ISBN : 978-3-319-46604-0.

[81] Pegah KHOSRAVI et al., « Deep learning enables robust assessment and selection of human blastocysts after in vitro fertilization », in : *npj Digital Medicine* 2.*1* (2019), p. 21, ISSN : 2398-6352, DOI : 10.1038/s41746-019-0096-y.

[82] Been KIM et al., « Interpretability Beyond Feature Attribution : Quantitative Testing with Concept Activation Vectors (TCAV) », in : *ICML*, 2018.

[83] Narine KOKHLIKYAN et al., *Captum A unified and generic model interpretability library for PyTorch*, 2020, arXiv : 2009.07896 [cs.LG].

[84] Jonathan KRAUSE et al., « 3D Object Representations for Fine-Grained Categorization », in : *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013.

[85] Alex KRIZHEVSKY, Ilya SUTSKEVER et Geoffrey E HINTON, « ImageNet Classification with Deep Convolutional Neural Networks », in : *Advances in Neural Information Processing Systems*, sous la dir. de F. PEREIRA et al., t. 25, Curran Associates, Inc., 2012.

[86] Isaac LAGE et al., *An Evaluation of the Human-Interpretability of Explanation*, 2019, DOI : 10.48550/ARXIV.1902.00006, URL : https://arxiv.org/abs/1902.00006.

[87] Tingfung LAU et al., « Embryo staging with weakly-supervised region selection and dynamically-decoded predictions », in : *CoRR* abs/1904.04419 (2019), arXiv : 1904.04419.

[88] Brian D. LEAHY et al., « Automated Measurements of Key Morphological Features of Human Embryos for IVF », in : *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, sous la dir. d'Anne L. MARTEL et al., Cham : Springer International Publishing, 2020, p. 25-35.

[89] Wei LI, Xiatian ZHU et Shaogang GONG, « Harmonious Attention Network for Person Re-Identification », in : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, juin 2018.

[90] Xueting LI et al., « Self-supervised Single-view 3D Reconstruction via Semantic Consistency », in : *ECCV*, 2020.

[91] Qiuyue LIAO et al., « Development of deep learning algorithms for predicting blastocyst formation and quality by time-lapse monitoring », in : *Communications Biology* 4.*1* (mars 2021), p. 415, ISSN : 2399-3642, DOI : 10.1038/s42003-021-01937-1.

[92] T. LIN, A. ROYCHOWDHURY et S. MAJI, « Bilinear CNN Models for Fine-Grained Visual Recognition », in : *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, p. 1449-1457.

[93] Tsung-Yu LIN, Aruni ROYCHOWDHURY et Subhransu MAJI, « Bilinear CNN Models for Fine-Grained Visual Recognition », in : *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, Washington, DC, USA : IEEE Computer Society, 2015, p. 1449-1457, ISBN : 978-1-4673-8391-2, DOI : 10.1109/ICCV.2015.170.

[94] Xiao LIU et al., « Fully Convolutional Attention Networks for Fine-Grained Recognition », in : *arXiv Computer Vision and Pattern Recognition* (2016).

[95] Yang LIU et Jeffrey HEER, « Somewhere Over the Rainbow : An Empirical Assessment of Quantitative Colormaps », in : CHI '18, Montreal QC, Canada : Association for Computing Machinery, 2018, p. 1-12, ISBN : 9781450356206, DOI : 10.1145/3173574.3174172, URL : https://doi.org/10.1145/3173574.3174172.

[96] Yaoyao LIU, Bernt SCHIELE et Qianru SUN, « An Ensemble of Epoch-Wise Empirical Bayes for Few-Shot Learning », in : *Computer Vision – ECCV 2020*, sous la dir. d'Andrea VEDALDI et al., Cham : Springer International Publishing, 2020, p. 404-421, ISBN : 978-3-030-58517-4.

[97] Z. LIU et al., « Multi-Task Deep Learning With Dynamic Programming for Embryo Early Development Stage Classification From Time-Lapse Videos », in : *IEEE Access* 7 (2019), p. 122153-122163, DOI : 10.1109/ACCESS.2019.2937765.

[98] Scott M LUNDBERG et Su-In LEE, « A Unified Approach to Interpreting Model Predictions », in : *Advances in Neural Information Processing Systems*, sous la dir. d'I. GUYON et al., t. 30, Curran Associates, Inc., 2017.

[99] Scott M LUNDBERG et Su-In LEE, « A unified approach to interpreting model predictions », in : *Proceedings of the 31st international conference on neural information processing systems*, 2017, p. 4768-4777.

[100] Laurens van der MAATEN et Geoffrey HINTON, « Visualizing Data using t-SNE », in : *Journal of Machine Learning Research* 9.*86* (2008), p. 2579-2605.

[101] S. MAJI et al., *Fine-Grained Visual Classification of Aircraft*, rapp. tech., 2013, arXiv : 1306.5151 [cs-cv].

[102] Luis MARTÍNEZ-GRANADOS et al., « Inter-laboratory agreement on embryo classification and clinical decision : Conventional morphological assessment vs. time lapse », eng, in : *PloS one* 12.*8* (août 2017), 28841654[pmid], e0183328-e0183328, ISSN : 1932-6203, DOI : 10.1371/journal.pone.0183328.

[103] Leland MCINNES, John HEALY et James MELVILLE, *UMAP : Uniform Manifold Approximation and Projection for Dimension Reduction*, 2020, arXiv : 1802.03426 [stat.ML].

[104] Ben MILDENHALL et al., « NeRF : Representing Scenes as Neural Radiance Fields for View Synthesis », in : *ECCV*, 2020.

[105] Margaret MITCHELL et al., « Model Cards for Model Reporting », in : *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, Atlanta, GA, USA : Association for Computing Machinery, 2019, p. 220-229, ISBN : 9781450361255, DOI : 10.1145/3287560.3287596, URL : https://doi.org/10.1145/3287560.3287596.

[106] Menaka NARAYANAN et al., *How do Humans Understand Explanations from Machine Learning Systems ? An Evaluation of the Human-Interpretability of Explanation*, 2018, DOI : 10.48550/ARXIV.1802.00682, URL : https://arxiv.org/abs/1802.00682.

[107] Meike NAUTA, Ron van BREE et Christin SEIFERT, *Neural Prototype Trees for Interpretable Fine-grained Image Recognition*, 2021, arXiv : 2012.02046 [cs.CV].

[108] Meike NAUTA et al., « This Looks Like That, Because ... Explaining Prototypes for Interpretable Image Recognition », in : *Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, Cham : Springer International Publishing, 2021, p. 441-456, ISBN : 978-3-030-93736-2.

[109] Nathan H NG et al., *Predicting Embryo Morphokinetics in Videos with Late Fusion Nets and Dynamic Decoders*, 2018.

[110] Jeremy NIXON et al., « Measuring Calibration in Deep Learning. », in : *CVPR Workshops*, t. 2, 7, 2019.

[111] A PAPATHEODOROU et al., « P-197 Successful implementation of an end-to-end artificial intelligence (AI) platform in a busy IVF clinic : A prospective observational study », in : *Human Reproduction* 37.*Supplement_1* (juin 2022), deac107.190, ISSN : 0268-1161, DOI : 10.1093/humrep/deac107.190, eprint : https://academic.

oup.com/humrep/article-pdf/37/Supplement_1/deac107.190/44307255/deac107.190.pdf.

[112] Richard J. PAULSON et al., « Time-lapse imaging : clearly useful to both laboratory personnel and patient outcomes versus just because we can doesn't mean we should », in : *Fertility and Sterility* 109.*4* (avr. 2018), p. 584-591, ISSN : 0015-0282, DOI : 10.1016/j.fertnstert.2018.01.042.

[113] F. PEDREGOSA et al., « Scikit-learn : Machine Learning in Python », in : *Journal of Machine Learning Research* 12 (2011), p. 2825-2830.

[114] Yuxin PENG, Xiangteng HE et Junjie ZHAO, « Object-Part Attention Model for Fine-Grained Image Classification », in : *IEEE Transactions on Image Processing* 27.*3* (mars 2018), p. 1487-1500, ISSN : 1941-0042, DOI : 10.1109/tip.2017.2774041.

[115] Vitali PETSIUK, Abir DAS et Kate SAENKO, *RISE Randomized Input Sampling for Explanation of Black-box Models*, 2018, arXiv : 1806.07421 [cs.CV].

[116] Nick A. PHILLIPS et al., *CheXphoto : 10,000+ Photos and Transformations of Chest X-rays for Benchmarking Deep Learning Robustness*, 2020, arXiv : 2007.06199 [eess.IV].

[117] PUBMEDDEV et Pribenszky C. al et et, *Time-lapse culture with morphokinetic embryo selection improves pregnancy and live birth chances and reduces early pregnancy loss : a meta-analysis. - PubMed - NCBI*, en, (visité le 12/02/2020).

[118] R. M. RAD et al., « Blastomere Cell Counting and Centroid Localization in Microscopic Images of Human Embryo », in : *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, août 2018, p. 1-6, DOI : 10.1109/MMSP.2018.8547107.

[119] Yongming RAO et al., « Counterfactual Attention Learning for Fine-Grained Visual Categorization and Re-identification », in : *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, p. 1005-1014, DOI : 10.1109/ICCV48922.2021.00106.

[120] Mengye REN et al., *Meta-Learning for Semi-Supervised Few-Shot Classification*, 2018, arXiv : 1803.00676 [cs.LG].

[121] Marco Tulio RIBEIRO, Sameer SINGH et Carlos GUESTRIN, « "Why Should I Trust You ?" : Explaining the Predictions of Any Classifier », in : *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, San Francisco, California, USA : Association for Computing Machinery, 2016, p. 1135-1144, ISBN : 9781450342322, DOI : 10.1145/2939672.2939778, URL : https://doi.org/10.1145/2939672.2939778.

[122] Olga RUSSAKOVSKY et al., « ImageNet Large Scale Visual Recognition Challenge », in : *International Journal of Computer Vision (IJCV)* 115.*3* (2015), p. 211-252, DOI : 10.1007/s11263-015-0816-y.

[123] Dawid RYMARCZYK et al., *Interpretable Image Classification with Differentiable Prototypes Assignment*, 2021, DOI : 10.48550/ARXIV.2112.02902, URL : https://arxiv.org/abs/2112.02902.

[124] Yuki SAWADA et al., « Artificial intelligence with Attention Branch Network and deep learning can predict live births by using time-lapse imaging of embryos after in vitro fertilisation », in : *Reproductive BioMedicine Online* (2021), ISSN : 1472-6483, DOI : https://doi.org/10.1016/j.rbmo.2021.05.002.

[125] Yuki SAWADA et al., « Evaluation of artificial intelligence using time-lapse images of IVF embryos to predict live birth », in : *Reproductive BioMedicine Online* 43.*5* (2021), p. 843-852, ISSN : 1472-6483, DOI : https://doi.org/10.1016/j.rbmo.2021.05.002.

[126] Lars SCHMARJE et al., « 2D and 3D Segmentation of Uncertain Local Collagen Fiber Orientations in SHG Microscopy », in : *Pattern Recognition*, sous la dir. de Gernot A. FINK, Simone FRINTROP et Xiaoyi JIANG, Cham : Springer International Publishing, 2019, p. 374-386, ISBN : 978-3-030-33676-9.

[127] R. R. SELVARAJU et al., « Grad-CAM : Visual Explanations from Deep Networks via Gradient-Based Localization », in : *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, p. 618-626.

[128] Junghoon SEO et al., « Noise-adding Methods of Saliency Map as Series of Higher Order Partial Derivative », in : *CoRR* abs/1806.03000 (2018), arXiv : 1806.03000, URL : http://arxiv.org/abs/1806.03000.

[129] Yaroslav SHMULEV et Mikhail BELYAEV, « Predicting Conversion of Mild Cognitive Impairments to Alzheimer's Disease and Exploring Impact of Neuroimaging », in : *Graphs in Biomedical Image Analysis and Integrating Medical Imaging and Non-Imaging Modalities*, sous la dir. de Danail STOYANOV et al., Cham : Springer International Publishing, 2018, p. 83-91, ISBN : 978-3-030-00689-1.

[130] Avanti SHRIKUMAR, Peyton GREENSIDE et Anshul KUNDAJE, « Learning Important Features through Propagating Activation Differences », in : ICML'17, Sydney, NSW, Australia : JMLR.org, 2017, p. 3145-3153.

[131] Julio SILVA-RODRÍGUEZ et al., « Predicting the Success of Blastocyst Implantation from Morphokinetic Parameters Estimated through CNNs and Sum of Absolute Differences », in : *2019 27th European Signal Processing Conference (EUSIPCO)*, 2019, p. 1-5, DOI : 10.23919/EUSIPCO.2019.8902520.

[132] David H. SILVER et al., « Data-Driven Prediction of Embryo Implantation Probability Using IVF Time-lapse Imaging », in : *Medical Imaging with Deep Learning*, 2020, URL : https://openreview.net/forum?id=TujK1uTkTP.

[133] Gurmail SINGH et Kin-Choong YOW, « An Interpretable Deep Learning Model for Covid-19 Detection With Chest X-Ray Images », in : *IEEE Access* 9 (2021), p. 85198-85208, DOI : 10.1109/ACCESS.2021.3087583.

[134] Gurmail SINGH et Kin-Choong YOW, « These do not Look Like Those : An Interpretable Deep Learning Model for Image Recognition », in : *IEEE Access* 9 (2021), p. 41482-41493, DOI : 10.1109/ACCESS.2021.3064838.

[135] Daniel SMILKOV et al., *SmoothGrad removing noise by adding noise*, 2017, arXiv : 1706.03825 [cs.LG].

[136] Jost Tobias SPRINGENBERG et al., « Striving for Simplicity : The All Convolutional Net », in : *CoRR* abs/1412.6806 (2015).

[137] Nitish SRIVASTAVA et al., « Dropout : A Simple Way to Prevent Neural Networks from Overfitting », in : *Journal of Machine Learning Research* 15 (2014), p. 1929-1958.

[138] J. STAAL et al., « Ridge-based vessel segmentation in color images of the retina », in : *IEEE Transactions on Medical Imaging* 23.*4* (2004), p. 501-509, DOI : 10.1109/TMI.2004.825627.

[139] Ashleigh STORR et al., « Inter-observer and intra-observer agreement between embryologists during selection of a single Day 5 embryo for transfer : a multicenter study », in : *Human Reproduction* 32.*2* (jan. 2017), p. 307-314, ISSN : 0268-1161, DOI : 10.1093/humrep/dew330.

[140] Guolei SUN et al., *Fine-grained Recognition Accounting for Subtle Differences between Similar Classes*, 2019, arXiv : 1912.06842 [cs.CV].

[141] Ke SUN et al., « Deep High-Resolution Representation Learning for Human Pose Estimation », in : *CoRR* abs/1902.09212 (2019), arXiv : 1902.09212.

[142] Mukund SUNDARARAJAN, Ankur TALY et Qiqi YAN, « Axiomatic Attribution for Deep Networks », in : *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, Sydney, NSW, Australia : JMLR.org, 2017, p. 3319-3328.

[143] D TRAN et al., « Deep learning as a predictive tool for fetal heart pregnancy following time-lapse incubation and blastocyst transfer », en, in : *Hum Reprod* 34.*6* (juin 2019), p. 1011-1018.

[144] Chun-Hua TSAI et Peter BRUSILOVSKY, « Evaluating Visual Explanations for Similarity-Based Recommendations : User Perception and Performance », in : New York, NY, USA : Association for Computing Machinery, 2019, p. 22-30, ISBN : 9781450360210.

[145] S UENO et al., « O-220 An annotation-free embryo scoring system (iDAScore®) based on deep learning shows high performance for pregnancy prediction after single-vitrified blastocyst transfer », in : *Human Reproduction* 36.*Supplement_1* (août 2021), deab128.044, ISSN : 0268-1161, DOI : 10.1093/humrep/deab128.044, eprint : https://academic.oup.com/humrep/article-pdf/36/Supplement_1/deab128.044/39735652/deab128.044.pdf.

[146] Rémi VALLÉE et al., « Influence of expertise on human and machine visual attention in a medical image classification task », in : *European Conference on Visual Perception*, online, France, août 2021, URL : https://hal.archives-ouvertes.fr/hal-03379161.

[147] Jasper VAN DER WAA et al., « Evaluating XAI : A comparison of rule-based and example-based explanations », in : *Artificial Intelligence* 291 (2021), p. 103404, ISSN : 0004-3702, DOI : 10.1016/j.artint.2020.103404.

[148] Ashish VASWANI et al., « Attention is All you Need », in : *Advances in Neural Information Processing Systems*, sous la dir. d'I. GUYON et al., t. 30, Curran Associates, Inc., 2017.

[149] Lucinda L. VEECK et Nikica ZANINOVIC, *An Atlas of Human Blastocysts*, en, Library Catalog : www.crcpress.com, (visité le 11/03/2020).

[150] M VERMILYEA et al., « Development of an artificial intelligence-based assessment model for prediction of embryo viability using static images captured by optical light microscopy during IVF », in : *Human Reproduction* 35.*4* (avr. 2020), p. 770-784, ISSN : 0268-1161, DOI : 10.1093/humrep/deaa013, eprint : https://academic.oup.com/humrep/article-pdf/35/4/770/33149528/deaa013.pdf.

[151] A. VITERBI, « Error bounds for convolutional codes and an asymptotically optimum decoding algorithm », in : *IEEE Transactions on Information Theory* 13.*2* (1967), p. 260-269.

[152] C. WAH et al., *The Caltech-UCSD Birds-200-2011 Dataset*, rapp. tech. CNS-TR-2011-001, California Institute of Technology, 2011.

[153] D. WANG et al., « Multiple Granularity Descriptors for Fine-Grained Categorization », in : *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, p. 2399-2406.

[154] Fei WANG et al., « Residual Attention Network for Image Classification », in : *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, p. 6450-6458, DOI : 10.1109/CVPR.2017.683.

[155] H. WANG et al., « Score-CAM : Score-Weighted Visual Explanations for Convolutional Neural Networks », in : *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Los Alamitos, CA, USA : IEEE Computer Society, juin 2020, p. 111-119, DOI : 10.1109/CVPRW50498.2020.00020.

[156] Haofan WANG et al., « Score-CAM Score-weighted visual explanations for convolutional neural networks », in : *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, p. 24-25.

[157] Jiaqi WANG et al., « Interpretable Image Recognition by Constructing Transparent Embedding Space », in : *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, oct. 2021, p. 895-904.

[158] Xiaolong WANG et al., « Non-local Neural Networks », in : *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, p. 7794-7803, DOI : 10.1109/CVPR.2018.00813.

[159] Yu WANG, Farshid MOUSSAVI et Peter LORENZEN, « Automated Embryo Stage Classification in Time-Lapse Microscopy Video of Early Human Embryo Development », in : *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*, sous la dir. de Kensaku MORI et al., Berlin, Heidelberg : Springer Berlin Heidelberg, 2013, p. 460-467, ISBN : 978-3-642-40763-5.

[160] Sarah WIEGREFFE et Yuval PINTER, *Attention is not not Explanation*, 2019, DOI : 10.48550/ARXIV.1908.04626, URL : https://arxiv.org/abs/1908.04626.

[161] Tianjun XIAO et al., « The Application of Two-level Attention Models in Deep Convolutional Neural Network for Fine-grained Image Classification », in : (2015).

[162] Wenxiao XIAO, Zhengming DING et Hongfu LIU, *Learnable Visual Words for Interpretable Image Recognition*, 2022, DOI : 10.48550/ARXIV.2205.10724, URL : https://arxiv.org/abs/2205.10724.

[163] Xiang XIE et al., « Early Prediction of Blastocyst Development via Time-Lapse Video Analysis », in : *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022, p. 1-5, DOI : 10.1109/ISBI52829.2022.9761654.

[164] Xiang XIE et al., « Early Prediction of Blastocyst Development via Time-Lapse Video Analysis », in : *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022, p. 1-5, DOI : 10.1109/ISBI52829.2022.9761654.

[165] Han XU et al., *Adversarial Attacks and Defenses in Images, Graphs and Text : A Review*, 2019, DOI : 10.48550/ARXIV.1909.08072, URL : https://arxiv.org/abs/1909.08072.

[166] H K YELKE et al., « O-007 Simplifying the complexity of time-lapse decisions with AI : CHLOE (Fairtility) can automatically annotate morphokinetics and predict blastulation (at 30hpi), pregnancy and ongoing clinical pregnancy », in : *Human Reproduction* 37.*Supplement_1* (juin 2022), deac104.007, ISSN : 0268-1161, DOI : 10.1093/humrep/deac104.007, eprint : https://academic.oup.com/humrep/article-pdf/37/Supplement_1/deac104.007/44305633/deac104.007.pdf.

[167] Zixu YUAN et al., « Development and Validation of an Image-based Deep Learning Algorithm for Detection of Synchronous Peritoneal Carcinomatosis in Colorectal Cancer », eng, in : *Annals of surgery* (juill. 2020), ISSN : 0003-4932, DOI : 10.1097/sla.0000000000004229.

[168] Hang ZHANG et al., « ResNeSt : Split-Attention Networks », in : *arXiv preprint arXiv :2004.08955* (2020).

[169] Jize ZHANG, Bhavya KAILKHURA et T. Yong-Jin HAN, « Mix-n-Match : Ensemble and Compositional Methods for Uncertainty Calibration in Deep Learning », in : *Proceedings of the 37th International Conference on Machine Learning*, sous la dir. d'Hal Daumé III et Aarti SINGH, t. 119, Proceedings of Machine Learning Research, PMLR, 13–18 Jul 2020, p. 11117-11128.

[170] Quanshi ZHANG et al., « Interpreting CNN knowledge via an Explanatory Graph », in : (2018).

[171] Zhizheng ZHANG et al., « Relation-Aware Global Attention for Person Re-Identification », in : *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, juin 2020.

[172] H. ZHENG et al., « Learning Multi-attention Convolutional Neural Network for Fine-Grained Image Recognition », in : *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, p. 5219-5227.

[173] Heliang ZHENG et al., « Looking for the Devil in the Details : Learning Trilinear Attention Sampling Network for Fine-grained Image Recognition », in : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, p. 5012-5021.

[174] L. ZHENG et al., « Scalable Person Re-identification A Benchmark », in : *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, p. 1116-1124, DOI : 10.1109/ICCV.2015.133.

[175] Zhedong ZHENG et al., « Joint discriminative and generative learning for person re-identification », in : *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[176] Bolei ZHOU et al., « Learning Deep Features for Discriminative Localization », in : *CoRR* abs/1512.04150 (2015), arXiv : 1512.04150.

[177] Kaiyang ZHOU et al., *Omni-Scale Feature Learning for Person Re-Identification*, 2019, arXiv : 1905.00953 [cs.CV].

DOCTORAT BRETAGNE LOIRE / MATHSTIC

Nantes Université

Titre : Attention spatiale artificielle pour des modèles interprétables de qualité embryonaire

Mot clés : FIV ; interprétabilité ; attention ; explicabilité ; embryon ; time-lapse

Résumé : L'un des traitements les plus courants de l'infertilité est la fécondation in vitro (FIV). Cette procédure consiste notamment à cultiver des embryons en milieu contrôlé et à en évaluer la qualité après plusieurs jours de croissance. La technologie time-lapse permet un suivi continu des embryons et génère une grande quantité d'images qui a déjà été exploitée par des applications d'apprentissage profonds. Une limitation importante au développement de ces solutions est la nature opaque des modèles proposés qui pose des problèmes éthiques déjà soulevés par la communauté.

Nous avons développé une base de données annotées par plusieurs experts pour permettre à la communauté de comparer les algorithmes développés et d'arriver à un consensus. Pour rendre les décisions des réseaux plus transparentes et explicables, nous avons travaillé sur un nouveau mécanisme d'attention artificielle non-paramétrique (BR-NPA). Nous comparons cette proposition avec l'état de l'art de l'attention visuelle artificielle du point de vue de la fiabilité des cartes de saillance produites à l'aide de métriques objectives. Nous discutons des limites de ces métriques et proposons d'autres métriques complémentaires.

Ce travail montre l'intérêt des modèles d'attention spatiale pour améliorer l'interprétabilité des modèles d'apprentissages profonds, dans le but d'aider les biologistes travaillant dans le domaine de la FIV mais aussi tous les praticiens utilisant des modèles de classification d'images dans leur travail quotidien.

Title: Artificial spatial attention for interpretable deep models of embryonic quality

Keywords: FIV; interpretability; attention; explainability; embryo; time-lapse

Abstract: In vitro fertilization (IVF) is one of the most common treatments for infertility. This procedure involves growing embryos in a controlled environment and assessing their quality after several days of growth. Time-lapse technology allows continuous monitoring of embryos and generates a large number of images that have already been exploited by deep learning applications. A significant limitation to the development of these solutions is the opaque nature of the proposed models, which poses problems, notably ethical ones, already raised by the community.

We have developed a database, annotated by several experts, which we have made public in order to allow the community to compare the developed algorithms and reach a consensus. To make model decisions more transparent and explainable, we have worked on a new non-parametric artificial attention mechanism (BR-NPA). We compare this proposal with the state of the art in artificial visual attention in terms of the reliability of saliency maps generated using objective metrics. We discuss the limitations of these metrics and propose other complementary metrics.

This work shows the interest in spatial attention models to improve the interpretability of deep learning models, to help biologists working in the field of IVF, and all practitioners using image classification models in their daily work.