Ajout non supervisé d'une modalité pour la télédétection

Jean-Christophe BURNEL, Sebastien LEFEVRE, Luc COURTRAI

Université Bretagne Sud, UMR 6074 IRISA
Campus de Tohannic, 56000 Vannes, France
jean-christophe.burnel@irisa.fr

Résumé – L'apprentissage profond nécessite une grande quantité d'annotations qui ne sont pas toujours disponibles pour toutes combinaisons de modalités et pour une tâche donnée. Cet article explore la question de l'ajout d'une modalité à un modèle pré-entraîné sans avoir recours à des annotations supplémentaires. En utilisant l'auto-entraînement, une technique d'auto-étiquetage, nous proposons non seulement de générer des pseudo-annotations pour des données multimodales, mais également de générer une pseudo-modalité pour les données annotées. Nous évaluons cette méthode sur un jeu de données avec comme première modalité l'orthophotographie couleur et en seconde modalité le modèle numérique de surface associé. Nous montrons que notre méthode permet d'améliorer les performances d'un réseau, notamment pour les classes qui nécessitent l'utilisation des deux modalités.

Abstract – Deep learning requires a large amount of annotations, which are not always available for all modality combinations and for a given task. This article explores the question of adding a modality to a pre-trained model without resorting to new annotations. By using self-training, a self-labeling method, we propose to not only generate pseudo-annotations for multimodal data, but also to generate pseudo-modality for annotated data. We propose to test this method on a dataset with color orthophotography as the first modality and the associated digital surface model as the second modality. Our method helps to improve the performances of a network, in particular for classes requiring the use of two modalities.

1 Introduction et contexte

Les méthodes d'apprentissage automatique, et notamment celles d'apprentissage profond, ont montré leur efficacité [9, 8] dans beaucoup de domaines d'application. Cependant les réseaux de neurones requièrent un grand nombre d'exemples pour l'entraînement et, selon les entrées et la tâche considérées, les jeux de données ne sont pas toujours disponibles. Dans un processus d'apprentissage profond, un exemple est composé d'entrées, souvent une image couleur, ainsi que d'une vérité terrain (VT). Ces VT peuvent prendre la forme d'une étiquette par image, pour une classification, ou d'un label par pixel, pour la segmentation sémantique. Dans ce dernier cas, la phase d'annotation est souvent longue et fastidieuse.

Prenons comme exemple une application de segmentation sémantique en télédétection du milieu urbain. On peut dans un premier temps estimer que l'orthophotographie peut suffire et profiter de services comme ceux offerts par l'IGN pour disposer d'une première base de données. Ces données devront être annotées afin de pouvoir entraîner un réseau de neurones pour la tâche souhaitée. En fonction des résultats, il est probable que certaines classes d'intérêt ne soient pas facilement discriminables via l'orthophotographie uniquement. L'ajout d'une nouvelle modalité peut alors devenir intéressant, afin de conduire à un gain de performance sur des classes d'intérêt.

La segmentation sémantique est un problème important en vision par ordinateur [7] et notamment en télédétection [3]. Certains travaux proposent d'utiliser l'auto-entraînement pour améliorer les performances [12], mais uniquement en imagerie couleur. D'autres proposent des méthodes pour tirer avantage de plusieurs modalités [10], mais toujours de manière supervisée. L'auto-entraînement [5] est une technique d'auto-étiquetage déjà explorée en télédétection. Cette méthode consiste à produire des annotations supplémentaires pour un jeu de données non supervisé via l'utilisation d'un réseau pré-entraîné que l'on appelle réseau enseignant. Ces annotations sont ensuite réutilisées dans une nouvelle phase d'entraînement. Les travaux récents sur l'auto-entraînement mettent en avant que le pseudo-étiquetage brut ne suffit pas, car il induit un biais de confirmation [2]. Pour contrer cet effet, l'augmentation de données ainsi que le bruitage du modèle ont démontré leur efficacité [11]. Dans ce contexte, on peut assimiler l'ajout d'une modalité à un mécanisme d'augmentation, ce qui nous permet de proposer une solution par auto-entraînement originale dont nous démontrons l'intérêt dans cet article.

Dans ce travail, on s'intéresse à l'ajout d'une modalité afin d'augmenter les performances en segmentation sémantique d'un modèle, et ce sans avoir besoin de nouvelles annotations. À cette fin, nous proposons d'utiliser un jeu de données multimodales non supervisé. Nous produisons non seulement des pseudo-labels pour ces données, mais aussi une nouvelle modalité pour le jeu de données déjà annoté.

La suite de cet article est organisée de la manière suivante. Dans la section 2, nous détaillons les différentes étapes de notre méthode. La section 3 présente une expérience sur notre adaptation du jeu de données ISPRS Potsdam [1]. Finalement, la

section 4 conclut ce travail.

2 Méthode

Notre méthode s'appuie sur l'auto-entraînement. On détaillera dans un premier temps comment est mis en place le pseudo-étiquetage, puis l'ajout de la pseudo-modalité, et enfin nous verrons comment nous intégrons ces deux éléments pour obtenir notre jeu d'entraînement final. Pour cela nous avons besoin de deux jeux de données, un premier qui contiendra la première modalité ainsi que les étiquettes : $(\boldsymbol{x}^1_{sup}, \boldsymbol{y}_{sup})$. Et un second jeu qui contiendra des exemples avec les deux modalités que nous voulons utiliser : $(\boldsymbol{x}^1_{uns}, \boldsymbol{x}^2_{uns})$.

2.1 Pseudo-labels

La génération de pseudo-labels requiert de disposer d'un réseau enseignant pré-entraîné sur les mêmes tâches que la tâche finale. Dans notre cas, ce réseau enseignant devra être pré-entraîné sur une première modalité. Pour la génération des pseudo-labels, on utilise le modèle pré-existant pour obtenir des prédictions pour chaque donnée du jeu non supervisé, en utilisant uniquement la première modalité, soit :

$$p = \mathcal{C}^s_\theta(x) \tag{1}$$

avec $\mathcal{C}^s_\theta$ le modèle entraîné de manière supervisée de paramètres θ et $x \in \mathcal{M}_1$, c'est-à-dire la modalité initiale.

Plutôt que d'utiliser les prédictions brutes de l'enseignant, nous appliquons à celles-ci un filtre similaire à [12] consistant à ne garder que les prédictions supérieures à un certain seuil. Ces seuils sont définis en fonction de la probabilité d'appartenance à une classe telle que rapportée par l'enseignant. Ainsi, si on ne souhaite conserver que les n% des prédictions les plus élevées, on fixera le seuil à la valeur correspondante. Cela permet de conserver la même proportion d'annotations par classe présente dans le jeu supervisé.

Ce filtrage est formellement décrit par

$$S(\boldsymbol{p}) = \{y : (y \in \boldsymbol{p}) \wedge (\sum_{i=0}^{n} \max(0, y_i - \mathcal{T}_i) > 0)\} \tag{2}$$

avec n le nombre de classes et $\mathcal{T}$ les seuils.

Finalement nos pseudo-labels sont obtenus en appliquant

$$PL = S(\mathcal{C}^s_\theta(\boldsymbol{x}^1_{uns})). \tag{3}$$

2.2 Pseudo-modalité

En plus des pseudo-labels, nous proposons d'utiliser un modèle $\mathcal{C}^c_\psi$ appelé contributeur dont l'objectif est d'estimer la seconde modalité depuis la première :

$$\mathcal{C}^c_\psi : \mathcal{M}_1 \rightarrow \mathcal{M}_2. \tag{4}$$

Les modèles utilisés pour cette étape sont dépendants des modalités concernées. Dans certains cas, tels que l'estimation de profondeur [6], le contributeur pourra être un réseau existant issu de travaux antérieurs. Si la seconde modalité est plus exotique, il sera nécessaire de concevoir une solution spécifique.

2.3 Nouveau jeu de données

Nous disposons donc de pseudo-labels pour notre jeu non supervisé et de pseudo-modalités pour notre jeu supervisé. Nous pouvons alors concaténer les deux afin de produire un jeu de données final. Celui-ci est obtenu avec :

$$ds = (\boldsymbol{x}^1_{sup}, \mathcal{C}^c_\psi(\boldsymbol{x}^1_{sup}), \boldsymbol{y}_{sup}) \cdot (\boldsymbol{x}^1_{uns}, \boldsymbol{x}^2_{uns}, S(\mathcal{C}^s_\theta(\boldsymbol{x}^1_{uns})), \tag{5}$$

Dans le cas de la segmentation sémantique, le filtrage introduit une nouvelle classe sous forme de masque. La fonction de coût associée devra en tenir compte afin de ne pas provoquer d'erreur.

3 Expériences

3.1 Protocole expérimental

Nous proposons d'évaluer notre méthode sur le jeu de données ISPRS Potsdam [1]. Ce jeu de données a l'avantage de posséder plusieurs modalités, notamment l'orthophotographie couleur (RVB) ainsi que le modèle numérique de surface (MNS) associé. On propose de modifier ce jeu de données afin de l'adapter à notre problématique où nous avons besoin d'un jeu de données RVB avec annotations et d'un jeu de données RVB+MNS sans annotation. Pour ce faire, nous proposons le découpage illustré dans la figure 2. Chaque tuile est de taille 6000x6000 pixels, qui seront ensuite recoupés en patch de 600x600. On dimensionnera ces patchs en 512x512 en entrée du réseau.

On utilisera en enseignant et en étudiant des réseaux basés sur Adapnet et SSMA [10]. Cela permet d'avoir des réseaux similaires, avec comme encodeur un ResNet-50 dans le cas mono-modal, et une combinaison d'un ResNet-50 et d'un ResNet-18 dans le cas multi-modal. Dans les deux cas, après l'encodeur, on utilisera un bloc eASPP, version plus légère du module ASPP [4]. La différence est que, dans le cas multi-modal, les données sont fusionnées à la fin de chaque bloc ResNet, excepté le premier.

On montre avec le Tableau 1 l'évolution de l'exactitude des pseudo-labels en fonction du taux de sélection, moins on choisit de pseudo-labels moins les erreurs sont présentes. Dans notre cas on prendra une valeur de 20% permettant d'avoir une quantité de pseudo-labels suffisante.

3.2 Résultats

Les résultats quantitatifs sont donnés dans la table 2 avec la précision par pixel (accuracy) pour chaque classe, ainsi que la moyenne des scores d'intersection sur union (mIOU). On remarque que, pour les trois premières classes (surfaces imperméables, bâtiments et végétation basse), les résultats sont proches. Cependant, pour les deux suivantes (arbres et voitures), les performances sont bien meilleures avec l'étudiant. Finalement, la dernière classe (autre) est bien moins bonne avec l'étudiant. Ces résultats semblent indiqués que l'étudiant ne tire

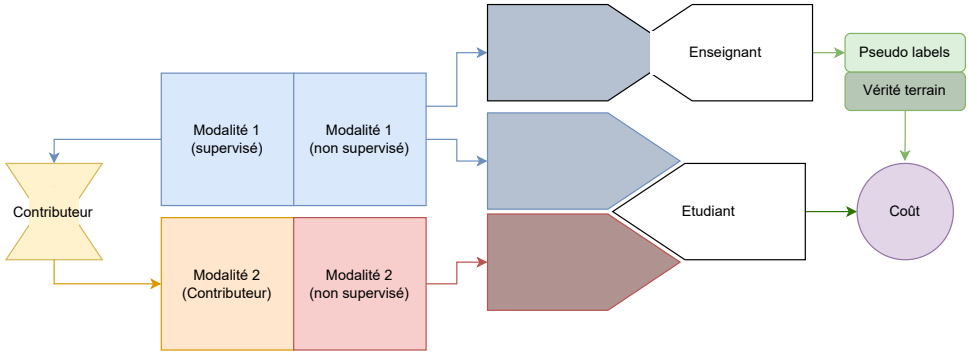


FIGURE 1 – Diagramme illustrant l'entraînement de l'étudiant. Les pseudo-labels sont générés grâce à l'enseignant, et la seconde modalité du jeu supervisé est générée avec le contributeur.

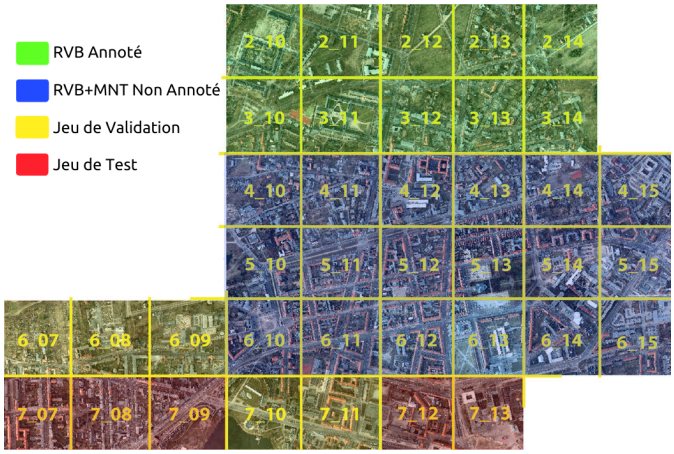


FIGURE 2 – Constitution de notre jeu de données à partir de ISPRS Potsdam [1].

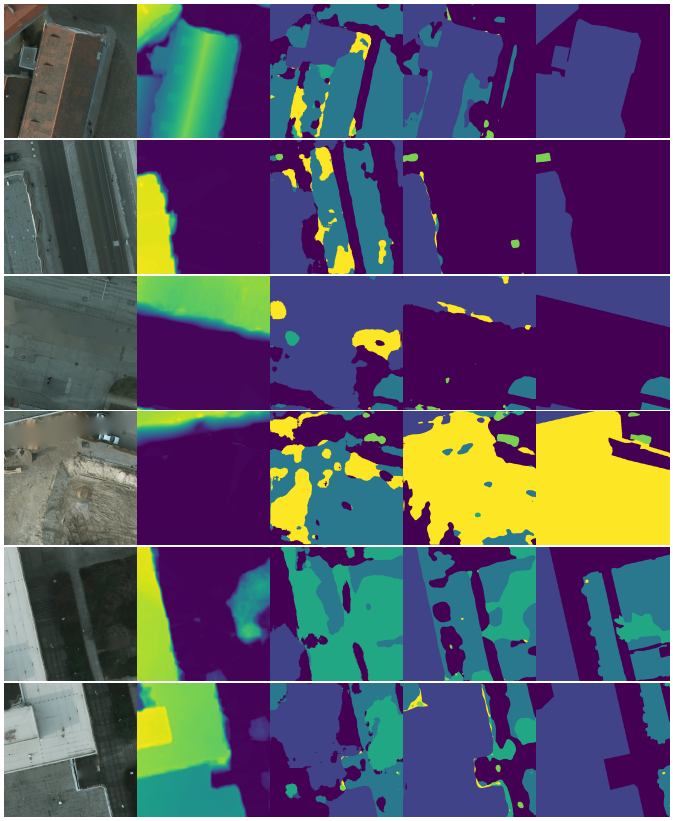
profit du MNS que pour deux classes alors que la classe bâtiment devrait aussi être mieux segmentée avec cet ajout.

Taux	0.9	0.8	0.6	0.4	0.2	0.1	0.05
Accuracy	84.58%	87.49%	91.91%	94.99%	97.34%	98.36%	99.11%

TABLE 1 – Exactitude des pseudos-labels en fonction du taux de sélection.

Ainsi, en complément de l'évaluation quantitative, nous proposons une évaluation qualitative des cartes de segmentation produites par les différents réseaux illustrées en figure 3. Globalement, les segmentations semblent de meilleure qualité avec l'étudiant, et on peut remarque qu'ici même les classes surfaces imperméables et bâtiments semble profiter de l'ajout de modalité. De plus, les prédictions de l'étudiant sur les deux dernières lignes semblent indiquer la présence d'arbres à la place de la végétation basse qui correspondrait avec l'élévation sur le MNS.

Pour comprendre les écarts entre les analyses qualitatives et quantitatives, il est intéressant de s'intéresser aux situations où l'étudiant est mis en échec, comme illustré dans la figure 4. Dans le premier exemple, où seule la classe eau est présente, le MNS est très bruité, et l'étudiant produit une carte



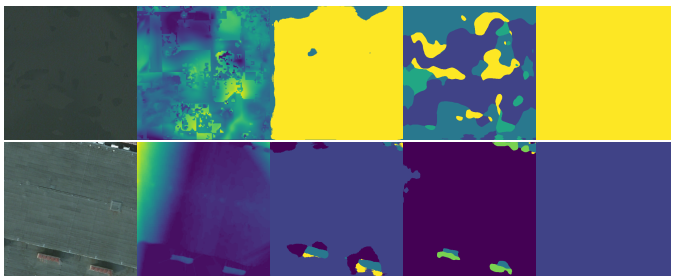
RVB MNS Enseignant Etudiant VT

FIGURE 3 – Comparaison des résultats entre l'enseignant mono-modal et l'étudiant multi-modal. On remarque que la qualité des segmentations est bien meilleure pour l'étudiant.

de segmentation également très bruitée. Cet exemple concerne la classe autre, classe où les performances de l'étudiant sont nettement moins bonnes que celle de l'enseignant. Dans le jeu de test, on remarque que cette situation est très présente, dégradant les performances globales de l'étudiant. Dans le deuxième exemple, une image d'un toit avec deux éléments se démarquant sur le MNS. On note également que, pour le toit, le MNS n'est pas régulier, alors qu'on aurait pu attendre une surface plane. L'étudiant confond les deux éléments avec des voitures

		Surfaces imperméables	Bâtiments	Végétation basse	Arbres	Voitures	Autres	Macro
Enseignant	Accuracy	78.62%	**81.98%**	**82.37%**	72.13%	70.90%	**34.63%**	70.1%
	IOU	0.67	0.77	0.51	0.57	**0.61**	**0.29**	56.96%
Etudiant	Accuracy	**80.89%**	81.82%	80.12%	**80.71%**	**86.02%**	29.68%	**73.21%**
	IOU	**0.72**	**0.78**	**0.54**	**0.60**	**0.61**	0.19	**57.25%**

TABLE 2 – Résultats expérimentaux sur ISPRS Potsdam.



RVB MNS Enseignant Etudiant VT

FIGURE 4 – Cas d'échec de l'étudiant comparé à l'enseignant.

et prédit également le toit comme une surface imperméable, allant de pair avec la présence de voitures. Ainsi on remarque que contrairement à l'enseignant, les erreurs sont ici plus généralisées : le réseau se trompe uniformément sur toute une surface, ce que nuit quantitativement aux résultats, mais donne plus de sens à l'analyse visuelle des résultats.

Pour conclure, l'ajout d'une modalité (même de manière non supervisée) permet donc bien d'obtenir des cartes de segmentation plus cohérentes. Pour certaines classes telles qu'arbres ou voitures, qui sont des classes plus rares et dont les instances sont plus petites (donc des classes plus difficiles), l'étudiant est plus précis, confirmant ainsi que le réseau tire bien parti du MNS pour la segmentation de ces classes. Cependant les cas d'échecs nous montrent également que le réseau se repose trop sur le MNS dans certains cas.

4 Conclusion

Nous avons montré qu'il était possible de profiter de l'ajout d'une nouvelle modalité sans avoir recours à une nouvelle phase d'annotation d'un jeu de données. Les expérimentations ont montré que cette méthode permettait d'obtenir de meilleures cartes de segmentation, la nouvelle modalité permettant de séparer plus facilement les classes, mais que cette dernière pouvait aussi introduire des erreurs de classification.

5 Remerciements

Ce travail est financé par la région Bretagne et le GIS BreTel via le projet doctoral ALTER, et par le FEAMP via le projet GAME OF TRAWLS. Ces travaux ont bénéficié d'un accès aux moyens de calcul de l'IDRIS au travers de l'allocation de ressources 2021-101245 attribuée par GENCI.

Références

[1] ISPRS Potsdam. https://www.isprs.org/education/benchmarks/UrbanSemLab/2d-sem-label-potsdam.aspx. Accessed : 2022-03-18.

[2] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *IJCNN*, 2020.

[3] N. Audebert, B. Le Saux, and S. Lefèvre. Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In *ACCV*, 2016.

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4) :834–848, 2017.

[5] I. Dópido, J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas Dias, and J. A. Benediktsson. Semisupervised self-learning for hyperspectral image classification. *IEEE TGRS*, 51(7) :4032–4044, 2013.

[6] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow. Digging into self-supervised monocular depth estimation. In *ICCV*, 2019.

[7] S. Hao, Y. Zhou, and Y. Guo. A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406 :302–321, 2020.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[10] A. Valada, R. Mohan, and W. Burgard. Self-supervised model adaptation for multimodal semantic segmentation. *IJCV*, 2019.

[11] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le. Self-training with noisy student improves imagenet classification. In *CVPR*, 2020.

[12] Y. Zou, Z. Yu, B. V. Kumar, and J. Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018.