Détection dense de changements par réseaux de neurones siamois

Rodrigo Caye Daudt, Bertrand Le Saux, Alexandre Boulch, Yann Gousseau

Détection dense de changements par réseaux de neurones siamois

Rodrigo Caye Daudt[1,2] Bertrand Le Saux[1] Alexandre Boulch[1] Yann Gousseau[2]

[1] ONERA - The French Aerospace Lab, FR-91761 Palaiseau, France
[2] Télécom ParisTech - LTCI, 46 rue Barrault, FR-75013 Paris, France
{rodrigo.daudt,bertrand.le_saux,alexandre.boulch}@onera.fr, yann.gousseau@telecom-paristech.fr

Résumé

Cet article présente d'une part une base de données publique de détection de changements urbains créée à partir d'images satellitaires multispectrales Sentinelle-2, et d'autre part des architectures de réseaux neuronaux convolutifs pour la détection de changements entre deux images. Les réseaux proposés sont des extensions siamoises d'architectures entièrement convolutives. Ils sont capables d'apprendre à détecter des changements en utilisant des paires d'images annotées en termes de changement, sans intervention humaine et sans post-traitement. Nous montrons leur efficacité tant sur les bases de données RVB de l'état de l'art que sur la nouvelle base multispectrale. En particulier, ces réseaux atteignent de meilleures performances que les méthodes précédemment proposées, tout en étant au moins 500 fois plus rapides que celles-ci.

Mots Clef

Détection de changements, apprentissage automatique, réseaux entièrement convolutifs, observation de la Terre.

Abstract

This paper presents convolutional neural network architectures which perform change detection using a pair of coregistered images. Most notably, we propose Siamese extensions of fully convolutional networks which use heuristics about the current problem to achieve the best results in our tests on two open change detection datasets, using both RGB and multispectral images. We show that our system is able to learn from scratch using annotated change detection images. Our architectures achieve better performance than previously proposed methods, while being at least 500 times faster than related systems. We also present a change detection dataset that was developed using Sentinel-2 images.

Keywords

Change detection, machine learning, fully convolutional networks, Earth observation.

1 Introduction

La Détection de Changements (DC) est un des principaux problèmes dans l'analyse d'images d'observation de la Terre. Son étude a une longue histoire et a évolué en parallèle des domaines du traitement de l'image et de la vision par ordinateur [1, 2]. Les systèmes de détection de changements ont pour but, pour une région géographique donnée, et à partir d'une paire ou d'une séquence d'images recalées prises à des dates différentes, d'étiqueter chaque pixel avec un label binaire. Une étiquette positive indique que la zone correspondant à ce pixel a changé entre les acquisitions. Bien que la définition de «changement» puisse varier d'une application à une autre, la DC est un problème de classification bien défini. Les changements peuvent concerner, par exemple, les changements de la végétation, l'expansion urbaine, la fonte des glaces polaires, etc. La DC est un outil particulièrement utile dans la production de cartes illustrant l'évolution de l'utilisation des sols, la couverture urbaine, la déforestation, etc.

Des programmes tels que Copernicus (satellites Sentinelles) et Landsat mettent à disposition de grandes quantités d'images d'observation de la Terre. Celles-ci peuvent être utilisés en conjonction avec les algorithmes d'apprentissage supervisé qui se sont développés ces dernières années, en particulier dans le domaine de l'analyse d'image. Dans le contexte de la détection de changements, il y a cependant un manque de grands jeux de données annotés, ce qui limite la complexité des modèles pouvant être utilisés.

Dans cet article, nous présentons plus en détails les travaux réalisés dans [3] et [4]. La première contribution est un jeux de données pour la détection de changements urbains, qui sera, à terme, mis à disposition de la communauté scientifique pour comparer différentes méthodes de détection de changements sur des images multispectrales. Nous présentons également deux architectures de réseaux de neurones convolutifs (en anglais *Convolutional Neural Network*, CNN) reposant sur des patchs (ou régions d'intérêt), ainsi que leurs extensions à des architectures dites entièrement convolutives (en anglais *Fully Convolutional Neural Network*, FCNN) qui effectuent la détection de changements sur des paires multi-temporelles d'images d'observation de la Terre. Ces architectures sont entraînées automatiquement de bout en bout (*end-to-end*) en utilisant uniquement les jeux de données de détection de changements disponibles. Les réseaux sont testés dans les cas Rouge-Vert-Bleu (RVB) et multispectral. Nos FCNN sont des réseaux encodeurs-décodeurs qui utilisent le concept de *skip*

connections introduit dans [5]. Nous proposons également pour la première fois deux architectures siamoises entièrement convolutives utilisant ces connexions.

Ce document est organisé comme suit. La section 2 propose un état de l'art sur les techniques de détection de changement et d'apprentissage. La partie 3 décrit plus en détails le jeux de données qui a été produit. La section 4 expose quant à elle en détail les méthodes proposées pour la détection de changements. Enfin, la partie 5 présente des comparaisons quantitatives et qualitatives avec des méthodes de détection de changements de la littérature.

2 État de l'art

Les travaux sur la détection de changements débutent avec les premières acquisitions d'images aériennes : voir [1, 2] pour une revue. Les techniques proposées ont suivi les tendances de la vision par ordinateur et de l'analyse d'images : dans un premier temps, les pixels ont été analysés directement en utilisant des techniques conçues manuellement ; plus tard, des descripteurs ont été utilisés en conjonction avec des techniques d'apprentissage standard [6] ; plus récemment, des techniques d'apprentissage automatique plus élaborées (*deep learning*) ont permis de résoudre de nombreux problèmes dans le domaine de l'analyse d'images, et cette évolution commence à s'étendre au problème de la détection de changements [7, 8, 9, 10, 11, 12, 3, 13].

En raison de la quantité limitée de données disponibles, la plupart de ces méthodes abordent le problème avec des techniques de *transfer learning*, en prenant comme point de départ des réseaux qui ont été entraînés sur de plus grandes bases de données disponibles pour d'autres problèmes. Ce type d'approche est limitante à bien des égards. En effet, elle suppose des similitudes entre ces bases de données et les données de détection de changements pertinentes. Par exemple, la plupart des réseaux à grande échelle ont été entraînés sur des images RVB et ne peuvent pas être transférés pour des images SAR ou multispectrales, ce qui est le cas de l'ensemble de données Sentinelles. Ces méthodes n'utilisent également pas l'entraînement *end-to-end*, qui tend pourtant à avoir de meilleurs résultats. Pour cette raison, nos travaux se concentrent sur des algorithmes capables d'apprendre uniquement à partir des données de détection de changements disponibles, et peuvent donc être appliqué des types de capteurs variés.

Utiliser des méthodes d'apprentissage pour comparer des images n'est pas une idée nouvelle. Les CNNs sont une famille d'algorithmes particulièrement adaptés à l'analyse d'images. Ils ont été appliqués dans différents contextes pour la comparaison de paires d'images [14, 15, 7]. Récemment, des architectures entièrement convolutives (FCNNs) ont été proposées pour des problèmes qui impliquent une prédiction dense, c'est-à-dire une prédiction au niveau du pixel [16, 5, 17]. Malgré l'obtention de très bons résultats sur d'autres problèmes d'observation de la Terre [18] et notamment leur supériorité par rapport aux approches basées sur des patchs ou des superpixels [19], ces techniques n'ont pas encore été appliquées à la DC à notre connaissance. Des architectures siamoises ont par ailleurs été proposées dans différents contextes dans le but de comparer des images [20, 15]. Cependant, à notre connaissance, le seul travail où un réseau siamois entièrement convolutif a été proposé auparavant est celui de Bertinetto et al. [17], utilisé pour résoudre le problème du suivi des objets dans les vidéos. Malgré l'obtention de bons résultats, l'architecture proposée dans ce travail est spécifique à ce problème et ne peut pas être transférée au cas des images satellitaires.

(a) Image 1 (b) Image 2 (c) Changements

FIGURE 1 – Exemple de carte de changement (c) générée manuellement par trois personnes différentes (représentées sur les 3 canaux de couleur) entre les images (a) et (b).

3 Dataset

Avec la croissance du nombre de programmes d'observation de la Terre tels que Copernicus et Landsat, de grandes quantités de données en accès libre peuvent être utilisées pour différentes applications. Les satellites Sentinelles-2 génèrent des séries temporelles d'images multispectrales avec des résolutions variant entre 10m et 60m par pixel, et ce à l'échelle du globe. Malgré l'abondance des données brutes, il y a peu d'ensembles de données étiquetés ouverts. Ceux-ci sont pourtant nécessaires pour développer des méthodes d'apprentissage supervisé. Les techniques de *deep learning* telles que les CNNs sont devenues populaires non seulement en raison de la croissance exponentielle de la puissance de calcul disponible, mais aussi en raison de la quantité de plus en plus grande de données annotées disponibles.

Outre l'apprentissage, l'absence de jeux de données d'évaluation ouverts rend difficile la comparaison quantitative des méthodes de DC. Nous présentons ici une nouvelle base de données pour la DC qui contient des cartes de changements annotées manuellement au niveau du pixel et qui vise à permettre cette évaluation. L'objectif de ce jeu de données pour la détection des changements urbains est de fournir un moyen ouvert et standardisé de comparer l'efficacité des différents algorithmes de DC proposés par la communauté scientifique, accessible à toute personne intéressée par le problème de DC. La base de données est dédiée à l'étude des zones urbaines. En effet, l'étiquetage n'identifie que la croissance et les changements urbains et ignore les changements naturels (par exemple, la croissance de la végétation ou les marées).

Le jeu de données fournit une norme de comparaison pour les algorithmes de DC et pour des données d'entrée va-

riées : un seul canal, image couleur ou multispectrales. Comme il contient des étiquettes de changements au sol pour chaque pixel sur chaque paire d'images, l'ensemble de données permet également d'appliquer des méthodes d'apprentissage supervisé élaborées spécifiquement pour le problème de la détection de changement.

Le jeu de données a été construit en utilisant des images des satellites Sentinelles-2, qui font partie du programme Copernicus. Les satellites capturent des images à diverses résolutions entre 10m et 60m par pixel dans 13 canaux entre l'ultraviolet et l'infrarouge à courte longueur d'onde (SWIR). Treize régions d'environ 500x500 pixels à 10m de résolution avec différents niveaux d'urbanisation où les changements urbains étaient visibles ont été choisies en France, en Espagne, en Italie et au Brésil. Les images ont été recadrées en fonction des coordonnées géographiques choisies, donnant lieu à 26 images pour chaque région, c'est-à-dire 13 canaux pour chacune des images dans la paire d'images. Ces images ont été téléchargées et recadrées à l'aide de la *toolbox* Medusa [1].

La grande variabilité des données brutes générées par Sentinelle-2 ne permet pas de créer des patchs d'image de façon entièrement automatisée. Les images téléchargées contiennent fréquemment de grandes régions entièrement noires et les images adéquates doivent être sélectionnées manuellement. De plus, pour la génération de cet ensemble de données, il était souhaitable d'obtenir des images avec peu ou pas de nuages présents dans l'image. Bien que l'interface *sentinelsat* permette un certain contrôle sur la quantité de nuages présents dans les images, cela nécessite néanmoins une vérification manuelle de chacune des images téléchargées pour s'assurer que les nuages ne sont pas trop présents.

Les étiquettes de Vérité Terrain (VT) par pixel ont été générées manuellement en comparant les images RVB pour chaque paire. Pour améliorer l'exactitude des résultats, la *toolbox* GEFolki [21] [2] a été utilisée pour recaler les images avec plus de précision que l'enregistrement effectué par le système Sentinel lui-même. Dans tous les cas, l'image la plus ancienne de la paire a été utilisée comme référence, et la plus récente a été transformée pour s'aligner parfaitement avec elle.

3.1 Défis et limites

Bien que la base de données soit très utile pour comparer méthodiquement différents algorithmes de DC et pour appliquer des techniques d'apprentissage supervisé à ce problème, il est important d'en souligner les limites. Tout d'abord, les images générées par le satellite Sentinel-2 ont une résolution relativement faible. Cette résolution permet de détecter l'apparition de grands bâtiments entre les images de la paire d'images. Des changements plus petits tels que l'apparition de petits bâtiments, l'extension de bâtiments existants ou l'ajout de voies à une route existante, par exemple, peuvent ne pas être évidents dans les images. Pour cette raison, même les cartes de changements générées manuellement diffèrent selon leur auteur.

La figure 1 montre la difficulté de définir et d'étiqueter avec précision les changements dans les paires d'images. Dans la figure 1(c), nous pouvons voir les cartes de changements effectuées par trois interprètes différents représentés dans des canaux de couleurs différents. Bien que plusieurs zones ont été annotées comme changements par les trois, il y a aussi beaucoup de désaccords. Cette différence vient de la difficulté de trouver une définition claire du changement qui couvre toutes les situations, même lorsque les images sont étiquetées manuellement. Certaines différences proviennent également de légères différences sur les emplacements exacts des limites des changements. Cela signifie que pour un tel ensemble de données, on ne peut pas espérer d'un algorithme pour atteindre des résultats parfaits, mais les résultats dans la section 5 montreront que cela n'invalide pas l'utilité de ce jeux de données.

Une approche alternative explorée pour la constitution de la vérité terrain consistait à utiliser des données OpenStreetMap de dates différentes pour générer les cartes de changements de manière automatisée. OpenStreetMap regroupe des données cartographiques ouvertes, et en comparant les cartes des dates des images dans les paires, il est théoriquement possible d'identifier les changements survenus dans la région. Cette approche s'est révélée infructueuse pour plusieurs raisons. Premièrement, la plupart des changements dans les cartes entre les deux dates étaient en fait dus à de nouveaux ajouts à la carte, et non pas des constructions nouvelles dans la période entre les dates où les images ont été prises. Deuxièmement, il n'est pas possible d'avoir beaucoup de précision sur la date des anciennes cartes : dans de nombreux cas, une seule carte était disponible pour chaque année avant 2017.

Enfin, le premier satellite Sentinelle-2 a été lancé en 2015 et, par conséquent, les données disponibles ne peuvent pas aller plus loin que juin 2015. Cela signifie que les changements contenus dans l'ensemble de données ont une distance temporelle d'au plus deux ans, et parfois moins que cela. Ainsi la quantité de changements est limitée et le nombre de pixels étiquetés comme non changement est beaucoup plus important que celui des pixels de changement.

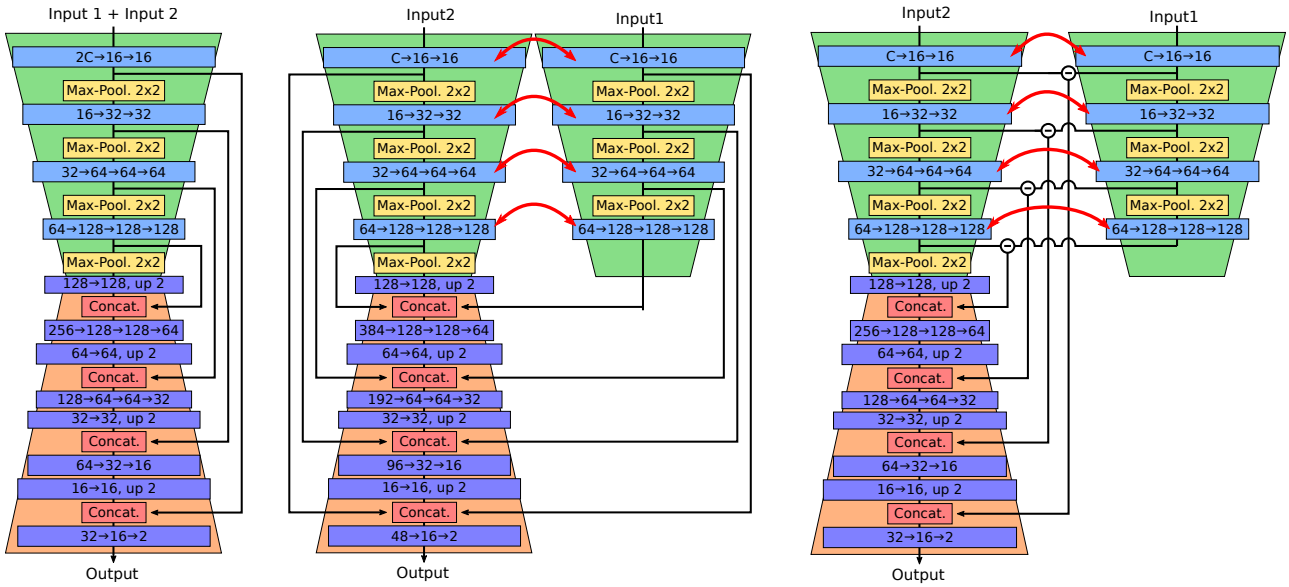
4 Méthodes

Les méthodes que nous présentons dans cet article peuvent être divisées en deux catégories : les réseaux qui classifient des patchs [3] et les réseaux entièrement convolutifs [4].

4.1 Réseaux par patch

Contrairement aux méthodes précédentes qui ne font qu'utiliser des CNN pour construire des images de différences qui sont ensuite seuillées, nos méthodes sont entraînées *end-to-end* pour classer un patch entre deux classes : changement et non-changement. Les patchs sont de taille

1. https://github.com/aboulch/medusa_tb
2. https://github.com/aplyer/gefolki



(a) FC-EF. (b)FC-Siam-conc. (c) FC-Siam-diff.

FIGURE 2 – Schémas des trois architectures FCNN proposées pour la détection de changements. Couleur des blocs : le bleu est la convolution, le jaune le *max pooling*, le rouge est la concaténation, le violet est la convolution transposée. Les flèches rouges illustrent les poids partagés entre les composantes siamoises du réseau.

15x15 pixels, et les réseaux classent l'étiquette du pixel central en fonction de son voisinage. Les réseaux devraient idéalement être capables d'apprendre à différencier les changements d'artificialisation et les changements naturels, étant donné que seuls les changements d'artificialisation sont classés comme des changements sur le jeux de données Sentinelle-2. Cette tâche est plus complexe qu'un simple calcul de différence entre les images, car elle implique une interprétation sémantique des images.

Nous proposons deux architectures CNN de classement par patchs. Ces réseaux prennent en entrée deux patchs 15x15xC, où C est le nombre de canaux de couleur. La sortie des réseaux pour chaque paire de patchs est une paire de valeurs qui sont une estimation de la probabilité que ce patch appartienne à chaque classe. En choisissant le maximum de ces deux valeurs, les réseaux prédisent si un changement s'est produit dans le pixel central du patch.

La première architecture proposée, appelée *Early Fusion* (EF), consiste à concaténer les deux paires d'images dans une première étape du réseau. L'entrée du réseau peut alors être vue comme un simple patch de 15x15x2C, qui est ensuite traité par une série de sept couches convolutives et deux couches entièrement connectées, où la dernière couche est une couche softmax avec les probabilités associées aux classes de changement et non changement.

La deuxième approche est un réseau siamois (Siam). L'idée est de traiter chacun des patchs en parallèle par deux branches de quatre couches convolutives avec des poids partagés, en concaténant les sorties et en utilisant deux couches entièrement connectées pour obtenir deux valeurs de sortie comme précédemment.

Une fois ces réseaux constitués, des cartes complètes de changement d'images peuvent être générées en classant les patchs des images de test individuellement. Pour accélérer ce processus, au lieu de prendre un patch centré en chaque pixel de l'image, un plus grand décalage (stride) a été utilisé pour extraire les patchs et un système de vote a été implémenté pour prédire les étiquettes de tous les pixels de l'image. Chaque patch classé vote sur l'étiquette de tous les pixels qu'il couvre en fonction des sorties du réseau, avec une pondération Gaussienne 2D centrée sur le pixel central du patch, ceci impliquant que le vote d'un patch est plus important pour les pixels plus proches de son centre.

4.2 Réseaux entièrement convolutifs

Les architectures entièrement convolutives proposées peuvent également être entraînées *end-to-end*, contrairement à la plupart des travaux récents sur la détection des changements. Ces architectures sont une évolution des réseaux présentées dans la section 4.1. L'évolution des architectures basées sur des patchs vers un schéma entièrement convolutif améliore la précision et la rapidité de l'inférence sans affecter de manière significative les temps d'entraînement. Ces réseaux sont également capables de traiter des entrées de tailles variables, à condition que la mémoire disponible soit suffisante, contrairement aux approches basées sur des patchs qui nécessitent des patchs de dimensions fixe, 15x15xC.

Pour étendre ces idées, nous avons utilisé le concept des *skip connections* qui ont été utilisées pour construire le réseau U-Net, destiné à la segmentation sémantique des images [5]. En bref, les *skip connections* sont des liaisons entre couches à la même échelle de sous-échantillonnage avant et après la partie de codage d'une architecture

encodeur-décodeur. L'idée est de compléter l'information plus abstraite et moins localisée présente dans les couches codées avec les détails spatiaux qui sont présents dans les premières couches du réseau, afin de produire une prédiction de classe précise avec des frontières bien localisées dans l'image de sortie.

La première architecture proposée est directement basée sur le modèle U-Net et sur l'architecture EF présentée précédemment ; elle est appelée *Full Convolutional Early Fusion* (FC-EF). Étant donné la quantité de données d'entraînement disponibles, le modèle U-Net original est trop complexe pour être directement appliqué à ce problème. Le FC-EF (fig. 2(a)) ne contient donc que quatre niveaux de max pooling et quatre de *upsamplings*, au lieu des cinq présents dans le modèle U-Net. Les couches de FC-EF sont également moins profondes que leurs équivalents U-Net. Comme dans le modèle EF basé sur un patch, l'entrée de ce réseau est la concaténation des deux images à comparer.

Les deux autres architectures proposées sont des extensions siamoises du modèle FC-EF. Pour ce faire, les couches de codage du réseau sont séparées en deux flux de structure égale, avec des poids partagés, comme dans un réseau siamois traditionnel. Chaque image est prise en entrée de l'un de ces flux. La différence entre les deux architectures siamoises réside uniquement dans la façon dont les *skip connections* sont effectuées. La première et plus intuitive consiste à concaténer les deux *skip connections* pendant les étapes de décodage, chacune provenant d'un flux d'encodage. Cette approche a été nommée *Fully Convolutional Siamese - Concatenation* (FC-Siam-conc, fig. 2(b)). Puisque dans la DC nous essayons de détecter les différences entre les deux images, cette heuristique a été utilisée pour combiner les *skip connections* d'une manière différente. Au lieu de concaténer les deux connexions à partir des flux de codage, nous concaténons plutôt la valeur absolue de leur différence. Cette approche est appelée *Fully Convolutional Siamese - Difference* (FC-Siam-diff, fig. 2(c)).

5 Résultats

Pour évaluer les méthodes proposées, nous avons utilisé deux jeux de données de détection de changement. Le premier est la base Onera Satellite Change Detection dataset [3] (OSCD)[3], et le second est le Air Change Dataset [22] (AC). AC contient des images aériennes RVB, tandis que OSCD contient des images satellites multispectrales. Les réseaux ont également été testés en utilisant uniquement les couches RVB de l'ensemble de données OSCD. Aux deux classes (changement et non changement) ont été assignés des poids inversement proportionnels au nombre d'exemples dans chaque classe. Les données disponibles ont été augmentées en utilisant les réflexions et rotations de multiples de 90 degrés des patchs d'entraînement. La technique de *dropout* a été utilisée pour éviter le surapprentissage pendant l'entraînement. Toutes les expériences ont été réalisées en utilisant le framework PyTorch et avec un GPU Nvidia GTX1070.

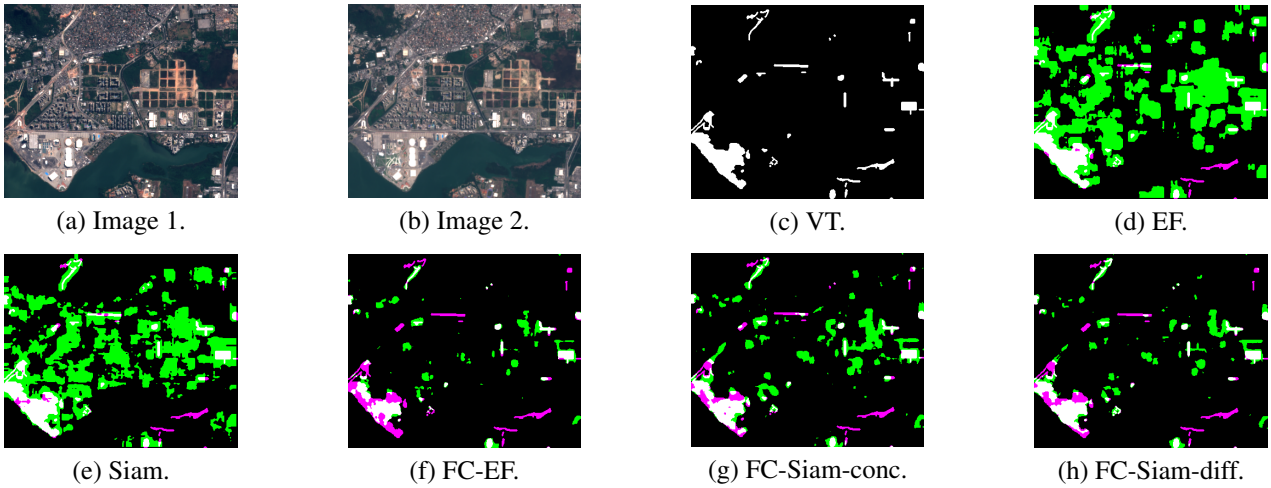
3. http://dase.ticinumaerospace.com / http://dase.grss-ieee.org

Data	Network	Prec.	Recall	Global	F1
OSCD-3 ch.	Siam. [3]	21.57	79.40	76.76	33.85
	EF [3]	21.56	**82.14**	83.63	34.15
	FC-EF	44.72	53.92	94.23	**48.89**
	FC-Siam-conc	42.89	47.77	94.07	45.20
	FC-Siam-diff	**49.81**	47.94	**94.86**	48.86
OSCD-13 ch.	Siam. [3]	24.16	**85.63**	85.37	37.69
	EF [3]	28.35	84.69	88.15	42.48
	FC-EF	**64.42**	50.97	**96.05**	56.91
	FC-Siam-conc	42.39	65.15	93.68	51.36
	FC-Siam-diff	57.84	57.99	95.68	**57.92**
Szada/1	DSCN [12]	41.2	57.4	-	47.9
	CXM [22]	36.5	58.4	-	44.9
	SCCN [9]	24.4	34.7	-	28.7
	FC-EF	**43.57**	62.65	**93.08**	51.40
	FC-Siam-conc	40.93	65.61	92.46	50.41
	FC-Siam-diff	41.38	**72.38**	92.40	**52.66**
Tiszadob/3	DSCN [12]	88.3	85.1	-	86.7
	CXM [22]	61.7	93.4	-	74.3
	SCCN [9]	**92.7**	79.8	-	85.8
	FC-EF	90.28	96.74	**97.66**	**93.40**
	FC-Siam-conc	72.07	**96.87**	93.04	82.65
	FC-Siam-diff	69.51	88.29	91.37	77.78

TABLE 1 – Évaluation quantitative sur les bases de données OSCD et Air Change.

Sur la base OSCD, nous séparons les données en jeux d'entraînement (14 images) et de test (10 images) comme proposé originalement au dataset. Pour l'ensemble de données AC, nous avons suivi la partition des données proposée dans [12] : le rectangle supérieur gauche de taille 748x448 des images Szada-1 et Tiszadob-3 a été extrait pour les tests, et le reste des données pour chaque emplacement a été utilisé pour l'apprentissage. Cela a permis une comparaison directe entre quatre algorithmes de DC. Chaque emplacement (Szada et Tiszadob) a été traité séparément en deux jeux de données différents, et les images nommées "Archieve" ont été ignorées, car elles ne contiennent qu'une seule paire d'images qui n'est pas suffisante pour entraîner les modèles présentés dans cet article.

Le tableau 1 contient l'évaluation quantitative des architectures de CD proposées, suivant les métriques standard. Pour la base de données AC, nos approches peuvent de plus être comparées aux 3 méthodes de l'état de l'art reflétant des approches sensiblement différentes : DSCN [12] utilisant des descripteurs appris de manière supervisée (CNN), CXM [22] qui utilise des modèles de Markov et des champs aléatoires conditionnels et SCCN [9], une méthode non supervisée (auto-encodeur). Les valeurs reportées ici sont celles de Zhan et al. dans [12]. La table contient les scores de précision, rappel et le score F1 du point de vue



(a) Image 1. (b) Image 2. (c) VT. (d) EF.

(e) Siam. (f) FC-EF. (g) FC-Siam-conc. (h) FC-Siam-diff.

FIGURE 3 – Résultats de test, images *rio* de la base OSCD, avec 13 canaux de couleur. Dans les images (d), (e), (f), (g) et (h), le blanc correspond au vrai positif, le noir au vrai négatif, les faux positifs sont en vert et les faux négatifs en magenta.

de la classe "changement", ainsi que la précision globale (pourcentage total des pixels bien classifiés) lorsqu'elle est disponible.

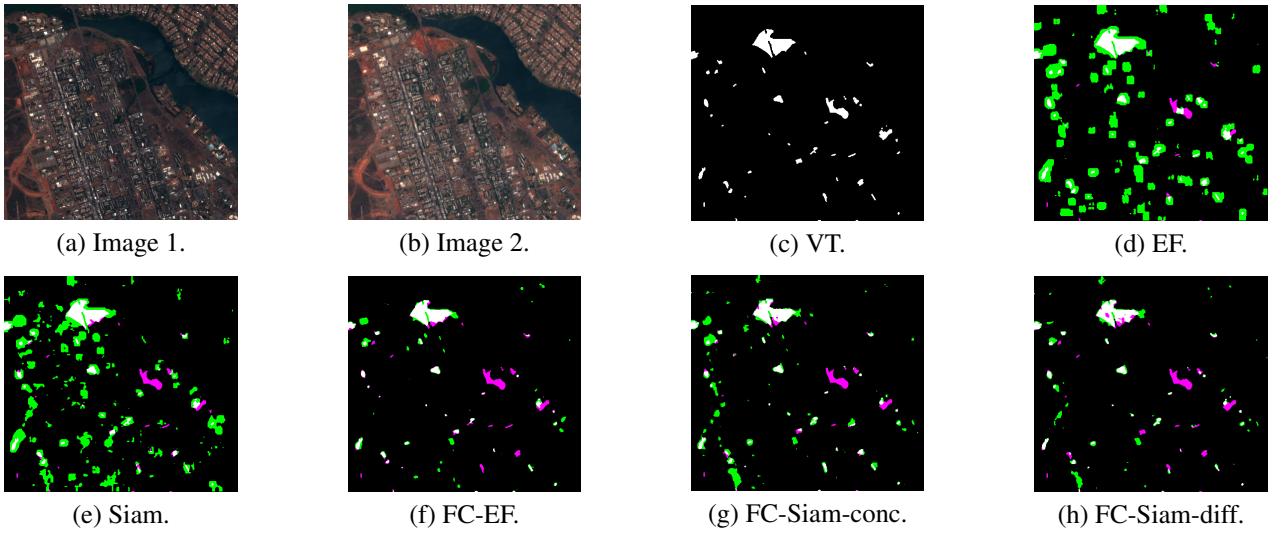
Les résultats relatifs à l'OSCD montrent que les méthodes entièrement convolutives proposées dans ce document dépassent de loin celles basées sur les patchs. Alors que les méthodes à patchs atteignent de bons taux de rappel, elles obtiennent de mauvaises précisions, ce qui réduit également le score F1. Le temps d'inférence des architectures entièrement convolutives était inférieur à 0.1 s par image pour tous nos cas de test, tandis que l'approche de vote par patch prenait plusieurs minutes pour prédire une carte de changement pour une image. Sur cet ensemble de données, le FC-Siam-diff a obtenu des scores F1 significativement meilleurs que toutes les autres méthodes proposées, ce qui tend à valider l'heuristique proposée. Cependant, les autres architectures entièrement convolutives sont également toujours nettement supérieures aux approches à patchs. Une illustration de nos résultats sur cet ensemble de données peut être trouvé dans les figures 3 et 4. Ces cartes de détection montrent l'apport de chaque méthode pour l'interprétation de données satellites : les cartes détections par des approches à patchs sont plus confuses car recouvertes de fausses alarmes.

Les résultats obtenus sur le jeux de données AC montrent également la supériorité de notre méthode par FCNN par rapport aux autres. Dans le cas de Szada/1, toutes les architectures FCNN proposées ont surpassé les autres méthodes utilisées pour le score F1, le FC-Siam-diff étant à nouveau la meilleure architecture. Pour le cas Tiszadob/3, le meilleur score F1 a été obtenu par notre architecture FC-EF, tandis que les autres architectures ont été surpassées par DSCN et SCCN. Encore une fois, le temps d'inférence des architectures entièrement convolutives était inférieur à 0.1 s, ce qui représente une accélération de plus de 500x par rapport au temps de traitement de 50 s revendiqué par Zhan et al. [12] pour la méthode SCCN sur un configuration très similaire, cette dernière utilisant un post-processing très coûteux. Les cartes de détection correspondantes sont présentées dans la figure 5.

Sur l'ensemble de ces tests, les architectures FC-Siam-diff et FC-EF semblent être les plus adaptées à la détection de changements. Ceci est, croyons-nous, dû à trois facteurs principaux qui rendent FC-Siam-diff particulièrement adapté à ce problème. Tout d'abord, des réseaux entièrement convolutifs ont été développés dans le but de traiter des problèmes de prédiction dense, tels que DC. Deuxièmement, l'architecture siamoise intègre dans la structure même du système une comparaison explicite entre deux images. La troisième raison est que les *skip connections* de différence guident explicitement le réseau pour comparer les différences entre les images, autrement dit, pour détecter les changements entre les deux images. On peut noter que l'architecture FC-EF, qui est plus générique, obtient elle aussi d'excellents résultats. Ses performances, légèrement en retrait vis-à-vis de FC-Siam-diff, pourraient s'expliquer par la difficulté d'apprendre les heuristiques utilisées dans le réseau siamois.

L'accélération significative de ces réseaux entièrement convolutifs sans perte de performance par rapport aux méthodes de DC précédentes est un pas vers un traitement efficace des flux massifs de données d'observation de la Terre qui sont disponibles via des programmes tels que Copernicus et Landsat. Ces programmes capturent de très grandes zones avec un taux de revisite élevé. Le déploiement de tels systèmes combiné avec des méthodes telles que celles proposées dans ce document permettrait un suivi mondial précis et rapide.

Ces résultats valident également la capacité de la base de données présentée à évaluer les algorithmes de détection des changements. Malgré quelques disparités entre les annotations manuelles et le faible nombre de paires d'images,



(a) Image 1. (b) Image 2. (c) VT. (d) EF.

(e) Siam. (f) FC-EF. (g) FC-Siam-conc. (h) FC-Siam-diff.

FIGURE 4 – Résultats de test sur *brasilia* de la base OS2UC, avec les 13 canaux de couleur. Dans les images (d), (e), (f), (g) et (h), le blanc correspond au vrai positif, le noir au vrai négatif, les faux positifs sont en vert et les faux négatifs en magenta.

l'apprentissage supervisé a été fait sans recourir à de l'apprentissage par transfert ou à l'utilisation d'une initialisation des réseaux avec des poids pré-entraînés.

6 Conclusion

Dans cet article, nous avons présenté la première base de données de détection de changements urbains Sentinelle-2 (à notre connaissance), qui sera mise à disposition de la communauté scientifique, ainsi que les méthodes utilisées pour sa génération, et les principaux défis rencontrés. Nous avons également présenté deux réseaux de neurones à base de patchs et trois réseaux entièrement convolutifs pour la prédiction dense. Ces cinq architectures ont été entraînées *end-to-end*. Les résultats obtenus avec nos méthodes surpassent l'état de l'art en matière de détection de changements, à la fois en précision et en vitesse d'inférence sans utilisation de post-traitement. Plus particulièrement, le paradigme encodeur-décodeur entièrement convolutif a été modifié en une architecture siamoise, tout en utilisant des *skip connections* pour améliorer la précision spatiale.

Une extension naturelle du travail présenté sur ce papier serait d'évaluer comment ces réseaux fonctionnent lorsqu'ils tentent de détecter des changements sémantiques. Il serait également intéressant de les tester avec d'autres modalités d'image (par exemple des images SAR), et de tenter de détecter des changements dans les séquences d'images. Il est également probable que ces réseaux profiteraient d'un entraînement sur des jeux de données plus grands, quand ceux-ci seront disponibles.

Références

[1] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *International journal of remote sensing*, vol. 10, no. 6, pp. 989–1003, 1989.

[2] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images : From pixel-based to object-based approaches," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 80, pp. 91–106, 2013.

[3] R. Caye Daudt, B. Le Saux, A. Boulch, and Y. Gousseau, "Urban change detection for multispectral earth observation using convolutional neural networks," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS'2018)*, July 2018.

[4] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *ICIP, submitted*, 2018.

[5] O. Ronneberger, P. Fischer, and T. Brox, "U-net : Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.

[6] B. Le Saux and H. Randrianarivo, "Urban change detection in sar images by interactive learning," in *Geoscience and Remote Sensing Symposium (IGARSS), 2013 IEEE International*, pp. 3990–3993, IEEE, 2013.

[7] S. Stent, R. Gherardi, B. Stenger, and R. Cipolla, "Detecting change for multi-view, long-term surface inspection.," in *BMVC*, pp. 127–1, 2015.

[8] A. M. El Amin, Q. Liu, and Y. Wang, "Convolutional neural network features based change detection in satellite images," in *First International Workshop on Pattern Recognition*, International Society for Optics and Photonics, 2016.

(a) Image 1.

(b) Image 2.

(c) Vérité terrain.

(d) DSCN.

(e) FC-EF.

(f) FC-Siam-conc.

(g) FC-Siam-diff.

FIGURE 5 – Comparaison entre les résultats obtenus par la méthode présentée dans [12] (d) et ceux décrits dans cet article (e-g) sur l'image Szada/1 de la base de données Air Change.

[9] J. Liu, M. Gong, K. Qin, and P. Zhang, "A deep convolutional coupling network for change detection based on heterogeneous optical and radar images," *IEEE transactions on neural networks and learning systems*, 2016.

[10] M. Gong, J. Zhao, J. Liu, Q. Miao, and L. Jiao, "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 1, pp. 125–138, 2016.

[11] A. M. El Amin, Q. Liu, and Y. Wang, "Zoom out cnns features for optical remote sensing change detection," in *Image, Vision and Computing (ICIVC), 2017 2nd International Conference on*, pp. 812–817, IEEE, 2017.

[12] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep siamese convolutional network for optical aerial images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1845–1849, 2017.

[13] L. Mou, X. Zhu, M. Vakalopoulou, K. Karantzalos, N. Paragios, B. Le Saux, G. Moser, and D. Tuia, "Multitemporal Very High Resolution From Space : Outcome of the 2016 IEEE GRSS Data Fusion Contest," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, pp. 3435–3447, June 2017.

[14] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 539–546, IEEE, 2005.

[15] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4353–4361, 2015.

[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[17] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *European conference on computer vision*, pp. 850–865, Springer, 2016.

[18] N. Audebert, B. Le Saux, and S. Lefèvre, "Beyond rgb : Very high resolution urban remote sensing with multimodal deep networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017.

[19] N. Audebert, B. Le Saux, and S. Lefèvre, "Segment-before-detect : Vehicle detection and classification through semantic segmentation of aerial images," *Remote Sensing*, vol. 9, no. 4, p. 368, 2017.

[20] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," in *Advances in Neural Information Processing Systems*, pp. 737–744, 1994.

[21] G. Brigot, E. Colin-Koeniguer, A. Plyer, and F. Janez, "Adaptation and evaluation of an optical flow method applied to coregistration of forest remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 7, pp. 2923–2939, 2016.

[22] C. Benedek and T. Szirányi, "Change detection in optical aerial images by a multilayer conditional mixed markov model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 10, pp. 3416–3430, 2009.