Master of Science HES-SO in Engineering

Av. de Provence 6

CH-1007 Lausanne

Master of Science HES-SO in Engineering

Orientation : Technologies de l’information et de la communication (TIC)

Une IA pour surveiller les engagements de zéro déforestation

Fait par

Romain Capocasale

Sous la direction de

Prof. Andres Perez-Uribe,

HEIG-VD

[Expert externe (Prénom et nom, titre de l’expert lors de la défence du projet d’appronfondissement)]

Lieu, HES-SO//Master, 20 ??

Accepté par la HES‑SO//Master (Suisse, Lausanne) sur proposition de

Prof. xxx, conseiller du projet d’approfondissement

[Xyz, Expert principal]

Lausanne, le … 20 ??

|  |  |
| --- | --- |
| Prof. xxx  Conseiller | Prof.  Responsable de la filière …. |

#### Dedicate

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed non risus. Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed, dolor. Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper.

# Table of Contents

[Table of Contents iv](#_Toc284593325)

[Acknowledgments v](#_Toc284593326)

[Abbreviations vii](#_Toc284593327)

[Abstract viii](#_Toc284593328)

[1. Introduction 1](#_Toc284593329)

[1.1. Aim of Study 1](#_Toc284593330)

[1.2. Scope and Limitation of Study 1](#_Toc284593331)

[2. Literature Review and Research Proposal 3](#_Toc284593332)

[2.1. State of the art 3](#_Toc284593333)

[2.1.1. Patent law 3](#_Toc284593334)

[3. Results 5](#_Toc284593335)

[3.1. Materials 5](#_Toc284593336)

[3.2. Products 5](#_Toc284593337)

[4. Discussion 7](#_Toc284593338)

[5. Experimental Part 9](#_Toc284593339)

[6. Conclusion 11](#_Toc284593340)

[7. References 13](#_Toc284593341)

[8. Appendices 15](#_Toc284593342)

# Acknowledgments

My sincere thanks goes to God almighty through whose inspiration and divine direction made it possible, for me, to accomplish and finally attained this academic level.

My inestimable appreciation goes to my family members for their moral advice and financial support throughout my study period …

# Abbreviations

|  |  |
| --- | --- |
| CNN | Convolutional Neural Network |
| CIAT | International Center for Tropical Agriculture |

# Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed non risus. Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed, dolor. Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper.

Duis arcu massa, scelerisque vitae, consequat in, pretium a, enim. Pellentesque congue. Ut in risus volutpat libero pharetra tempor. Cras vestibulum bibendum augue. Praesent egestas leo in pede. Praesent blandit odio eu enim. Pellentesque sed dui ut augue blandit sodales. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam nibh. Mauris ac mauris sed pede pellentesque fermentum. Maecenas adipiscing ante non diam sodales hendrerit. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Sed non risus. Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed, dolor. Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper. Duis arcu massa, scelerisque vitae, consequat in, pretium a, enim. Pellentesque congue. Ut in risus volutpat libero pharetra tempor. Cras vestibulum bibendum augue. Praesent egestas leo in pede. Praesent blandit odio eu enim. Pellentesque sed dui ut augue blandit sodales. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam nibh. Mauris ac mauris sed pede pellentesque fermentum. Maecenas adipiscing ante non diam sodales hendrerit.

Key Words: Fermentum, adipiscing, scelerisque, elementum, pellentesque, hendrerit, risus, tortor, dignissim, eleifend, blaendit, mauris.

https://www.mdpi.com/2072-4292/12/6/901/htm

https://hal.archives-ouvertes.fr/hal-03044231

# Introduction

Au Vietnam, de nombreuses entreprises ont signé des accords de zero-deforestation. Cependant ces entreprises ne savent pas si les produits qu’elle achètent proviennent d’une zone déforesté ou non.

En collaboration avec le centre de recherche sur l'Agriculture Tropicale (CIAT) au Vietnam, l’idée est de développer différents réseaux de neurone capable d’indiquer si une certaine zone a été déforesté et remplacé par des champs de café.

Les images utilisées proviennent la mission européenne Sentinel-2 et le CIAT fourni des annotations de champs de café dans le but d’entrainé des réseaux de neurones à détecter la présence ou non d’un champ de café.

De nombreux projets utilisent déjà des réseaux de neurones pour détecter la déforestation dans certaines zones. La grande majorité de ces projets utilisent pour ceci des architectures CNN standard et des images satellites. Au vu du grand nombre de réseau à disposition il peut être contreproductif de réentraîner son propre réseau de neurones (surtout si le nombre d'exemple labelisé est faible). La solution proposée dans ce projet est donc d'utiliser des modèles pré-éentraîné et d'effectuer du transfer learning.

Le transfer learning consiste à utiliser un réseau de neurones réentraîné sur une grande quantité d’image. Le modèle pré-entrainé a appris à reconnaitre les caractéristiques de bas niveau des images et uniquement les couches supérieures, les caractéristiques de hauts niveaux sont entrainées sur le jeu de données cible. Cette approche permet de :

* Gagner du temps lors de l’entrainement
* Diminuer le nombre de données labelisé nécessaire.
* Utiliser les caractéristiques de bas niveau connu.

## But de l’étude

Ce projet d’approfondissement se base sur travail de bachelor sur la même thématique [1]. La méthodologie du précédent travail consistait a entrainé les réseaux de neurones “From Scratch”. Cette solution demande du temps et un jeu de données de grande taille.

Le but de ce projet est d’utiliser des techniques de transfert learning pour effectuer la même tâche de classification que le travail de bachelor avec un modèle préentrainé.

Le modèle de transfert learning principalement utilisé est le résultat d’un travail de master [2]. Ce modèle a été entrainé sur le jeu de données Eurosat.

Les principales étapes du projet sont les suivants :

1. Mettre en place le pipeline de transformation des images satellite proposé dans les préccedent travaux à disposition.
2. Entrainement de réseaux de neurones avec des modèles préentrainés sur Eurosat trouvés sur Internet.
3. Entrainement de réseaux de neurones avec le modèle préentrainé proposé par le travail de Master
4. Entrainement de réseaux de neurones pour détecter les terrains de type culture contre les terrains de type non-culture.
5. Comparaison des résultats obtenu avec le réseau de neurones « From scratch » des précédents travaux.

## Scope and Limitation of Study

….

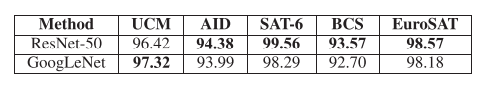
# Analyse documentaire et proposition de recherche

## État de l’art

Il existe plusieurs réseaux de neurones entrainés sur des images satellites dans le but de classifier différents types de terrains. C’est notamment le cas du projet Eurosat [3] qui propose un benchmark des scores obtenu sur des images satellites de paysage européen labelisé à la main. La comparaison a été effectué avec différentes architectures CNN.

Pour ce benchmark les 13 bandes des images satellite ont été utilisées. La taille des images est de 64x64.

Voici un aperçu des l’accuracy obtenues avec le jeu de données Eurosat avec train-test split de 80/20:



*Figure 1 : P. Helber & al. results on their two best-performing fine-tuned CNNs[1, p. 6]*

On constate que le réseau de neurones qui obtient le meilleur résultat est **ResNet-50**.

# Etat du projet

Ce chapitre permet de donner une vue d’ensemble sur les deux projets qui définisse la base de ce travail.

## Réseau de neuronne from Scratch

Le projet sur lequel se base ce travail [1] propose différentes architectures de réseaux de neurones entrainé de manière “from scratch”.

Différentes méthodes d’entrainements ont été utilisé comme :

* Variation du nombre de bande utilisé
* Combinaison de certaine bande entre elle
* Prise en compte de la saison d’acquisition des images
* Utilisation de différentes sources d’images Satellite (Sentinel 2A, 1C, 1)

Voici un tableau récapitulatif des méthodologies utilisé dans ce travail.

|  |  |
| --- | --- |
| **Images utilisées** | Sentinel2A, Sentinel1C et Sentinel1 |
| **Taille des images** | 15 x 15 |
| **Nombre de bande** | Entre 1 et 4 bandes |
| **Normalisation** | Min-Max normalisation |

*Tableau 1 : Paramètres travaux J.Rod*

Voici un tableau des trois meilleurs modèles obtenus avec les images Sentinel 1C:

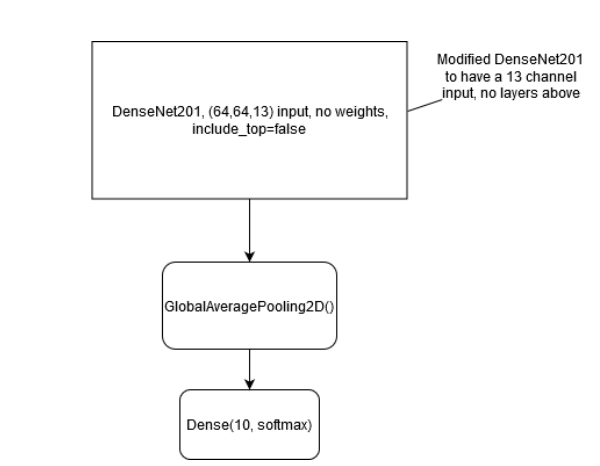
|  |  |  |
| --- | --- | --- |
| **Bandes utilisées** | **Saison** | **F-Score** |
| Bleu, Vert, Rouge et Infra-Rouge | Hiver | 76.21% |
| Bleu, Rouge et Infra-Rouge | Automne | 76.10% |
| Bleu, Vert, Rouge et Infra-Rouge | Automne | 75.99% |

*Tableau 2 : Scores modèles Sentinel 1C J.Rod*

## Modèle pré-entrainé DenseNet Eurosat

Le principale modèle qui est utilisé pour le transfer learning provient d’un travail de master d’un ancien étudiant [2]. Ce travail consistait entre autres à prédire à partir d’une image satellite le type de terrain (Industriel, Forêt, Rivière, …). Ce modèle a été entrainé à partir du jeu de données Eurosat [3].

Voici une représentation de l’architecture utilisée. Le modèle se base sur une architecture DensetNet modifié pour prendre en entrée les 13 bandes des images Eurosat.



*Figure 2: Architecture réseau de neurone E.Ransom*

Le modèle a obtenu les résultats suivants :

|  |  |
| --- | --- |
| **Accuracy** | **F-Score** |
| 97.58 % | 97.0 % |

*Tableau 3 : Score modèle E.Ransome*

Voici un tableau récapitulatif de la méthodologie utilisé dans ce travail.

|  |  |
| --- | --- |
| **Images utilisées** | Eurosat |
| **Taille des images** | 64 x 64 |
| **Nombre de bande** | 13 bandes |
| **Normalisation** | Z-norm |

*Tableau 4 : Paramètres travaux E.Ransome*

Au vu des résultats obtenu, on constate qu’il est très intéressant d’utiliser ce modèle pour effectuer du transfer learning. Les caractéristiques de bas niveau apprises sur les images pourront donc être utilisée pour ce problème.

# Solutions développées

Ce chapitre présente des différentes solutions développées pour ce projet.

## Prétraitement des images

Dans le but de rester le plus proche possible des images Eurosat et du modèle utilisé pour le transfer learning les points suivants ont été respectés.

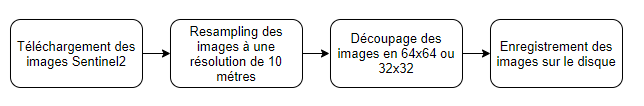
* Utilisation des images Sentinel 1C
* Taille des images en 64x64
* Normalisation z-norm

Tout d’abord un script a été mis en place pour télécharger les images Sentinel 2 à l’aide du package Python SentinelSat.

Étant donné que les bandes des images Sentinel2 n’ont pas toutes la même taille de pixel[[1]](#footnote-1), il faut donc effectuer une interpolation bicubique pour que toutes les bandes ai une taille de pixel de 10 mètres comme indiqué dans le travail de E.Ransom [2] (Page 7, Chapitre4.3).

Une fois toutes les bandes de même taille, les images originales de taille 10980x10980 sont découpé en image de taille de taille 64x64. Il est à noter que comme 10980 n’est pas divisible par 64, une bande de 36 pixels sur la droite et en bas de l’image n’est pas utilisée. Les images découpées sont ensuite localisées et labelisé et fonction des fichiers Shapefile fournit par le CIAT. Il est à noter que pour certaine expérience des images de taille 32x32 sont utilisées.

La dernière étape consiste a enregistré les images sur le disque en fonction du label qui lui a été attribué. Ces différentes étapes de prétraitement ont été repris du travail de J. Rod [1].



*Figure 3 : Étapes de préprocessing*

Ces différentes opérations se trouvent dans le notebook : *ImageProcessing.ipynb*.

## Générateur et augmentation de données

Étant donné la taille importante des image un générateur a été mis en place. Un générateur permet de charger les images et de les fournir au réseau de neurones à chaque epochs à la volée. Il est à noter que le DataGenerator proposé par Keras[[2]](#footnote-2) ne peut pas être utilisé pour ce projet car il ne gère que les images entre 1 et 4 bandes. Un générateur personnalisé prenant en compte les 13 bandes est mis en place.

Pour augmenter la taille du jeu de données des techniques d’augmentation de données sont mise en place. Les transformations sur les images sont appliquées dans le générateur et effectué à la volée avant d’être transmise au réseau de neurone. Les transformations suivantes sont appliquées aux images :

* Rotation d’un angle de 90°, 180° ou 270°
* Inversion verticale ou horizontal des bits de l’image.

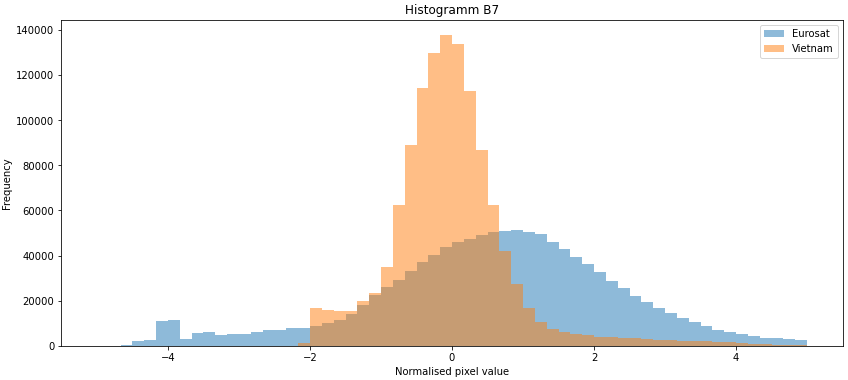
## Normalisation

La normalisation des images est appliquée également à la volée à l’aide du générateur. La méthode de normalisation appliqué sur chaque pixel de l’image est Z-norm.

Avec :

* la valeur normalisé du pixel (i, j) de la bande k
* le pixel original de coordonnées (i, j) de la bande k
* la moyenne de toutes les images Eurosat pour la bande k
* l’écart-type de toutes les images Eurosat pour la bande k

Cependant quand on applique la normalisation avec les paramètres calculés sur le jeu de données Eurosat on constate que les pixels ne sont pas bien centrés autour de 0. Ceci provient du fait que la moyenne et l’écart-type ont été calculé sur les images des paysages européen. Pour contrer cela, des expériences ont également été réalisées en utilisant la moyenne et l’écart-type des images Sentinel2 des paysages du Vietnam utilisé pour le projet.



*Figure 4 : Histogramme des pixels normalisés avec les paramètres Eurosat et Vietnam pour la bande B7*

On constate qu’avec les paramètres du Vietnam les pixels de l’image sont plus centrés autour de 0.

## RestNet-50 Tensorflow Hub

Comme première essaie d’architecture, un modèle ResNet-50 entrainé sur des images Eurosat[[3]](#footnote-3) a été récupéré sur Tensorflow hub. Ce modèle est alors utilisé pour effectuer du transfert learning. Il a noté que ce modèle utilise par défaut des images d’entrée de la forme 224x224x3 alors qu’idéalement on souhaite exploiter toutes les bandes des images satellite avec une résolution de 64x64 ou 32x32.

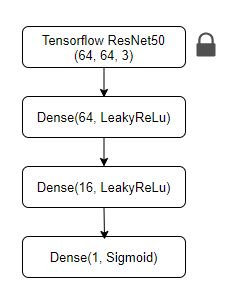
Ce modèle sera toutefois utilisé, étant donné qu’il est possible de préciser la taille des images d’entrée désire. De plus uniquement les 3 bandes RGB seront utilisées.

Récapitulatif de la couche Tensorflow Hub utilisée :

|  |  |
| --- | --- |
| **Images utilisées** | Eurosat |
| **Taille des images** | 224x224 |
| **Nombre de bande** | 3 bandes |

*Tableau 5 : Paramètres modèle Tensorflow Hub*

L’architecture utilisé pour cette expérience est la suivante :



*Figure 5 : Architecture modèle Tensorflow Hub*

Note :

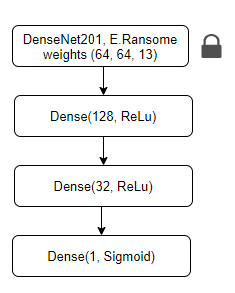
* Entre chaque couche exceptée la dernière, un dropout de 0.5 et une régularisation L2 sont appliqués.
* Les poids de la première couche du réseau sont gelés pendant la phase d’entrainement (représenté par le cadenas sur le schéma).

Ces différents modèles se trouvent dans le notebook : *TrainingTensorflowHub.ipynb*.

## DenseNet Transfer learning

Dans cette expérience, le modèle de transfer learning du chapitre 3.2 avec les **13** bandes est utilisé. Des images de taille 64x63 ou 32x32 ont été utilisés pour entrainer ce modèle.

L’architecture utilisé pour cette expérience est la suivante :



*Figure 6 : Architecture modèle transfer learning*

Note :

* Entre chaque couche exceptée la dernière, un dropout de 0.5 et une régularisation L2 sont appliqués.
* Pour la premier module les poids du modèle d’E.Ransome de 13 bandes sont utilisés.
* Les poids de la première couche du réseau sont gelés pendant la phase d’entrainement (représenté par le cadenas sur le schéma).

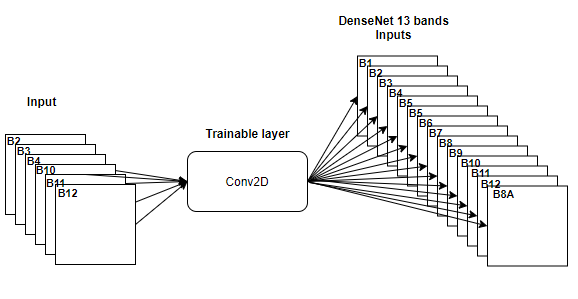
Ces différents modèles se trouvent dans le notebook : *TrainingTransferMulti.ipynb*.

### Variation du nombre de bande

D’après cet article Medium [6] il apparait qu’une amélioration des résultats se produit en utilisant différente combinaison de bande en non en utilisant automatiquement les 13 bandes.

Dans le but de réaliser ceci, l’architecture présentée précédemment est utilisée. Une couche de Convolution est ajoutée en entrée de ce réseau dans le but de modifier le nombre de bande utilisé en entrée. Cette couche est définie comme « trainable » pour que le réseau puisse effectuer de manière non supervisé une sélection des bandes. Le nombre de bande en entré est plus petit que le nombre de bande d’entré du réseau DenseNet pré-entrainé.

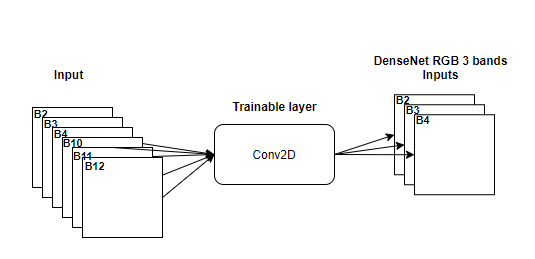
Le schéma ci-dessous présent un exemple d’architecture avec 6 bandes en entrée :



*Figure 7 : Exemple architecture couche “trainable” 13 bandes*

Des expériences alternatives ont été réalisé en utilisant le modèle de transfer learning pré-entrainé d’E.Ransome mais cette fois ci entrainé sur des images RGB (3 bandes). De cette manière, le nombre de bande en entré est plus grand que le nombre de bande d’entré du réseau DenseNet pré-entrainé.

Le schéma ci-dessous présent un exemple d’architecture avec 6 bandes en entrée :



*Figure 8 : Exemple architecture couche “trainable” RGB 3 bandes*

* Énumération des solution développées :
  + DenseNet 13 bandes :
    - Bande RGB (B2, B3, B4) + SWIR (B10, B11 et B12)
    - Bande haute résolution (B2, B3, B4, B5, B6, B7, B8, B11, B12, B8A)
  + DenseNet 3 bandes :
    - Bande RGB (B2, B3, B4) + SWIR (B10, B11 et B12)
    - Bande haute résolution (B2, B3, B4, B5, B6, B7, B8, B11, B12, B8A)
    - 13 Bandes

Ces différents modèles se trouvent dans le notebook : *BandVariation.ipynb.*

## Culture vs non-culture

Une expérience réalisée en parallèle avec les différentes méthodologies présentées ci-dessus est d’effectuer une classification pour différentier les paysages de cultures des paysages non-cultures.

Les classes considérées comme **culture** sont :

* Cacao
* Café
* Caoutchouc
* Poivre
* Thé
* Riz
* Banane

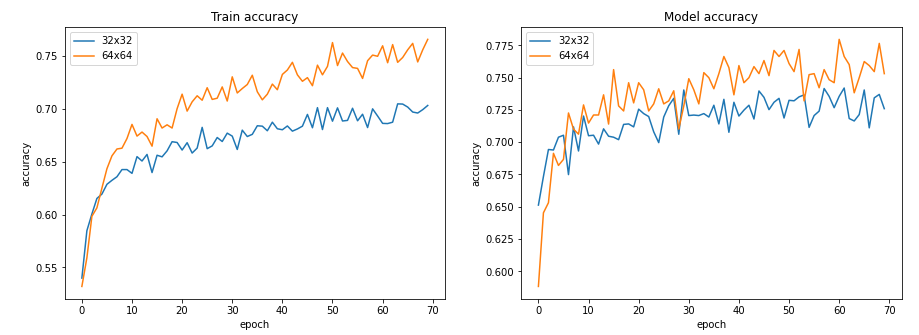
Les classes considérées comme **non-culture** sont :

* Végétation native
* Urbain
* Eau
* Forêt feuillue
* Arbre de pin

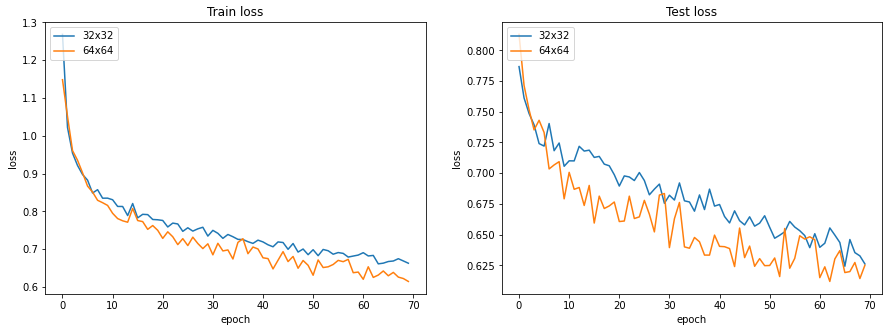
Les mêmes architectures de modèle que précédemment ont été utilisés.

# Résultats

## ResNet50 Tensorflow Hub



*Figure 9 : ResNet50 - Tensorflow Hub accuracy*

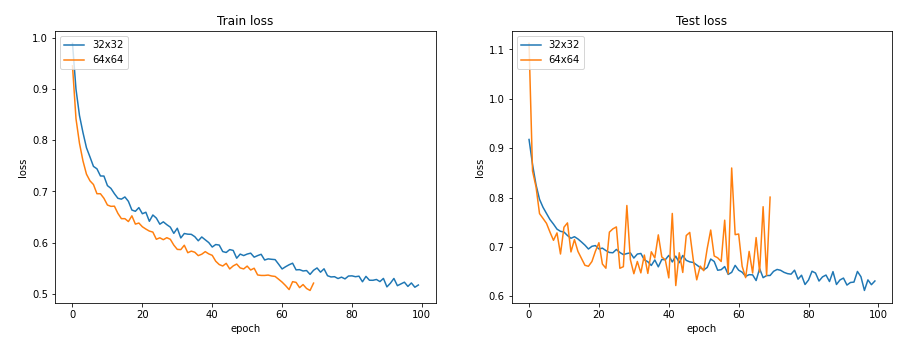


*Figure 10 : ResNet50 - Tensorflow Hub loss*

## DenseNet201 Transfer learning

# 

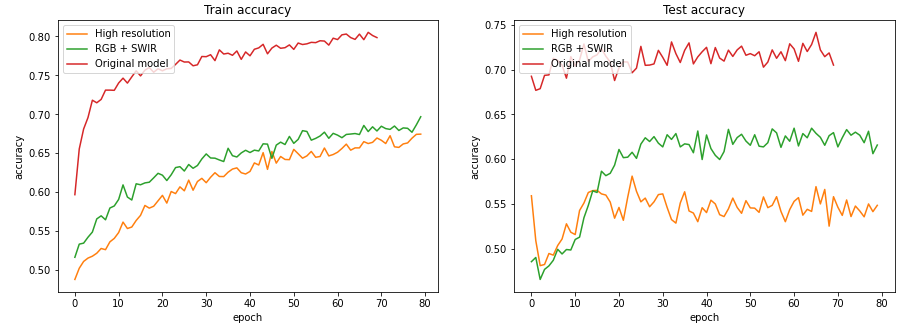
*Figure 11 : DenseNet201 – Transfer learning accuracy*



*Figure 12 : DenseNet201 – Transfer learning loss*

## Variation de bandes

### DenseNet 3 bandes



*Figure 13 : Variation de bandes – DenseNet 3 bandes accuracy*

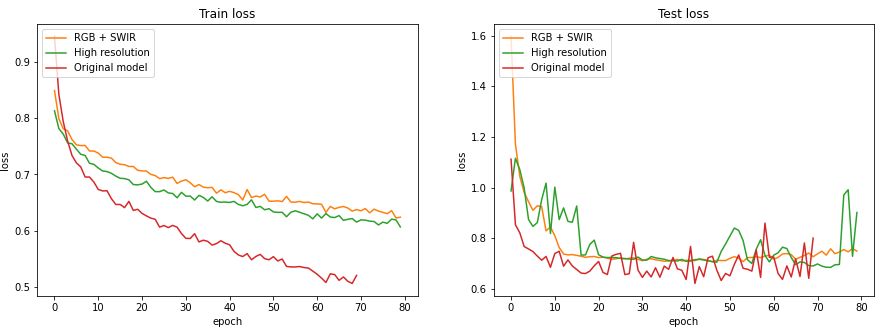
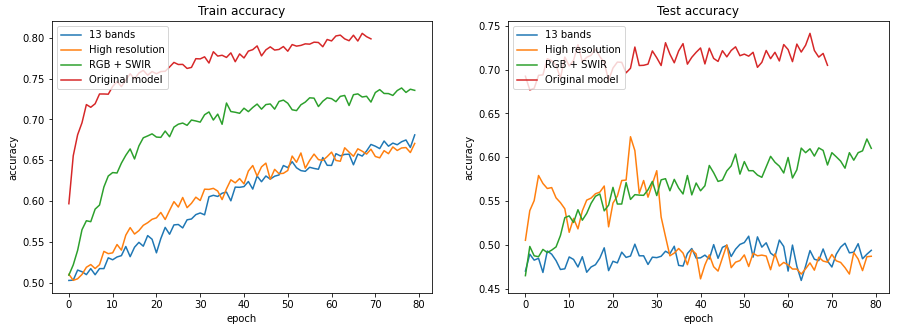
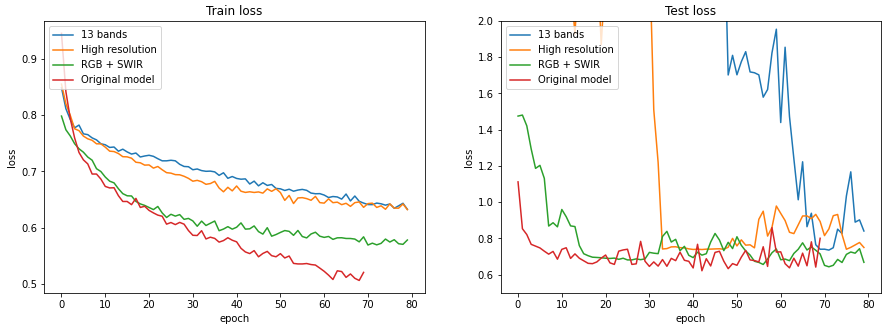


Figure 14 : *Variation de bandes – DenseNet 3 bandes loss*

### DenseNet 13 bandes



*Figure 15 : Variation de bandes – DenseNet 13 bandes accuracy*



*Figure 16 : Variation de bandes – DenseNet 13 bandes loss*

## Culture contre non-culture

# Discussion

# Experimental Part

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed non risus. Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed, dolor. Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper. Duis arcu massa, scelerisque vitae, consequat in, pretium a, enim. Pellentesque congue. Ut in risus volutpat libero pharetra tempor.

# Conclusion

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Sed non risus. Suspendisse lectus tortor, dignissim sit amet, adipiscing nec, ultricies sed, dolor. Cras elementum ultrices diam. Maecenas ligula massa, varius a, semper congue, euismod non, mi. Proin porttitor, orci nec nonummy molestie, enim est eleifend mi, non fermentum diam nisl sit amet erat. Duis semper. Duis arcu massa, scelerisque vitae, consequat in, pretium a, enim. Pellentesque congue. Ut in risus volutpat libero pharetra tempor.

Sion, February 15, 2011.

Capocasale Romain

# References

1. Rod Julien, L'intelligence artificielle au service de l'agriculture durable, 2020
2. Ransome Edward, Deep learning for automatic mapping from satellite imagery, 2020
3. Helber, Patrick and Bischke, Benjamin and Dengel, Andreas and Borth, Damian, 2019 Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing <https://arxiv.org/abs/1709.00029>
4. Géron Aurélien, Machine Learning avec Scikit-Learn (1ère édition). Paris, France : Dunod, (2017)
5. Géron Aurélien, Deep Learning avec TensorFlow (1ère édition). Paris, France : Dunod, (2017)
6. Abdishakur 2018, Land use/Land cover classification with Deep Learning, <https://link.medium.com/ncZaHz2YLeb>
7. [fchollet](https://twitter.com/fchollet), 2020Transfer learning & fine-tuning, <https://keras.io/guides/transfer_learning/>

# Appendices

List of the appendices

#### Appendix I

Package Python utilisé :

* rasterio
* cv2
* matplotlib
* numpy
* shapefile
* affine
* sklearn
* pyproj
* tensorflow
* tensorflow\_hub
* keras

Appendix II

Appendix III

1. https://developers.google.com/earth-engine/datasets/catalog/COPERNICUS\_S2 [↑](#footnote-ref-1)
2. https://keras.io/api/preprocessing/image/#imagedatagenerator-class [↑](#footnote-ref-2)
3. https://tfhub.dev/google/remote\_sensing/eurosat-resnet50/1 [↑](#footnote-ref-3)