

Recherche en Deep Learning sur la segmentation semantique d'image

Kerdoncuff Tanguy

Laboratoire Hubert Curien, UMR CNRS 5516

May 13, 2020

Table des matières

1 Segmentation sémantique

- Un exemple de segmentation
- Résolution du problème par le Machine Learning
- Exemple de segmentation par GridNet

2 Code en Pytorch de GridNet

3 Étude des fonctions de perte

- Fonction de coût et fonction de perte
- Minimisation direct de l'IoU
- Extension de Lovasz

Table des matières

1 Segmentation sémantique

- Un exemple de segmentation
- Résolution du problème par le Machine Learning
- Exemple de segmentation par GridNet

2 Code en Pytorch de GridNet

3 Étude des fonctions de perte

- Fonction de coût et fonction de perte
- Minimisation direct de l'IoU
- Extension de Lovasz

Un exemple de segmentation

Segmentation d'image

La segmentation d'image a pour objectif de donner la classe d'appartenance de chaque pixel d'une image. Les différentes classes étant définies au préalable.



Figure 1: Exemple d'image et la segmentation associée, tiré du jeu de données Cityscapes Dataset [1]

Résolution du problème par le Machine Learning

Une approche utilisant le Deep Learning

- Nécessite une grande base de donnée d'entraînement : Cityscapes Dataset (10 milliards de pixel segmenté manuellement)
- Nécessite une grande puissance de calcul : cluster/GPU



- N couples d'entraînement : images et vérités terrains associés $(X_0, Y_0 \dots X_{N-1}, Y_{N-1})$
- Un réseau de neurones f_θ dont les paramètres θ seront appris sur les couples d'entraînements. On prédit Y_n avec $f_\theta(X_n) = \hat{Y}_n$
- Modification des paramètres θ pour minimiser une fonction de perte choisie ($loss(Y_n, f_\theta(X_n))$)
- Test sur M couples : images et vérités terrains associés $(X_N, Y_N \dots X_{N+M-1}, Y_{N+M-1})$ avec une fonction de coût : $cout(Y_n, f_\theta(X_n))$

Les convolutions

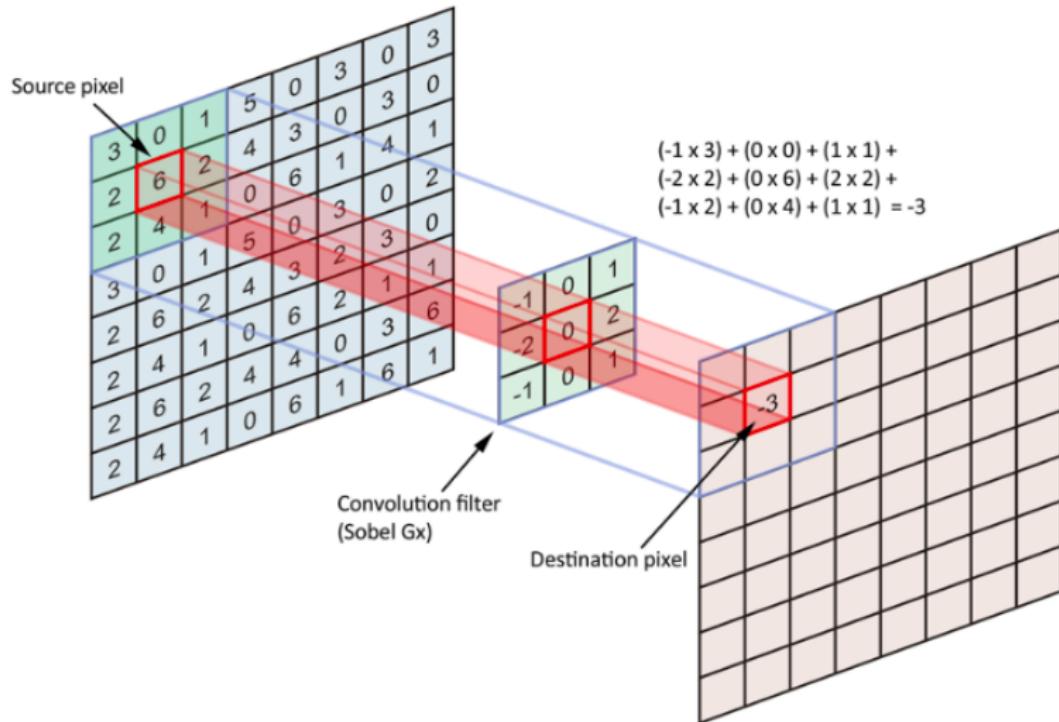


Figure 2: Schéma d'une image 8x8 par un filtre 3x3 avec calcul d'un pixel de sortie

Les convolutions

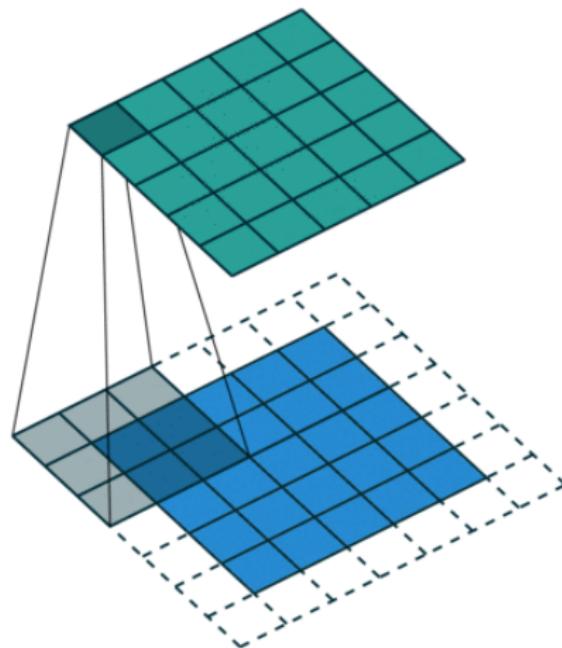


Figure 3: Schéma de convolution 3x3 avec conservation de dimension et mise en évidence du calcul du premier pixel

Les convolutions

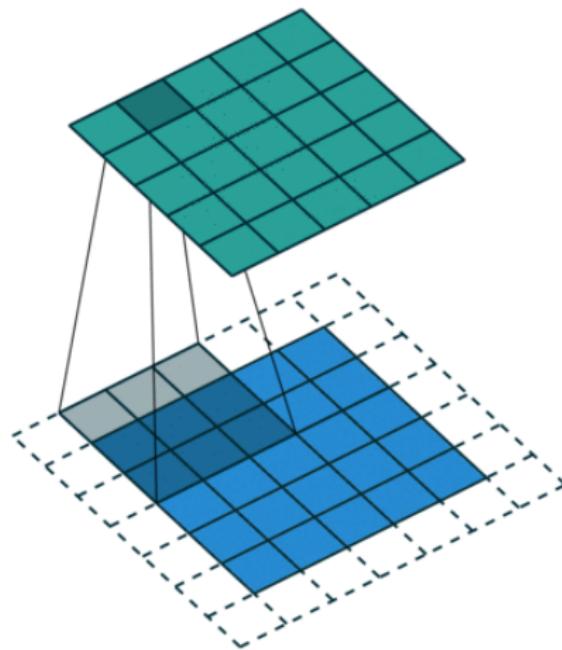


Figure 4: Schéma de convolution 3x3 avec conservation de dimension et mise en évidence du calcul du second pixel

Les convolutions

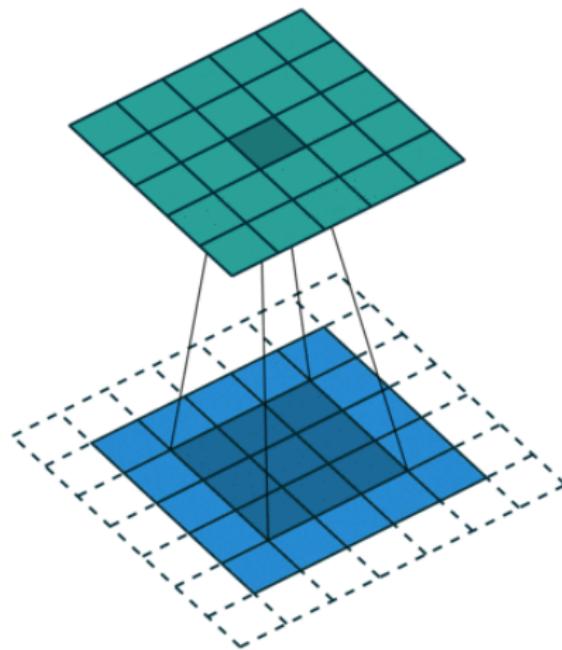


Figure 5: Schéma de convolution 3x3 avec conservation de dimension et mise en évidence du calcul du pixel du milieu

Les convolutions

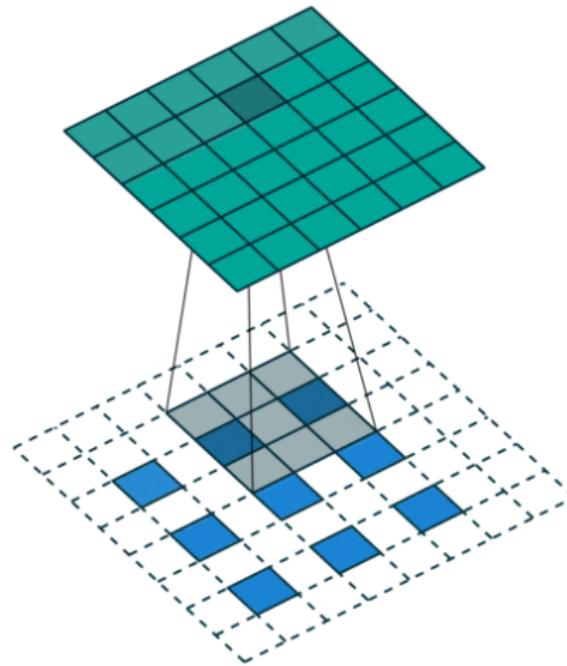


Figure 6: Schéma de convolution 3x3 avec augmentation de la dimension

Architecture originale : GridNet

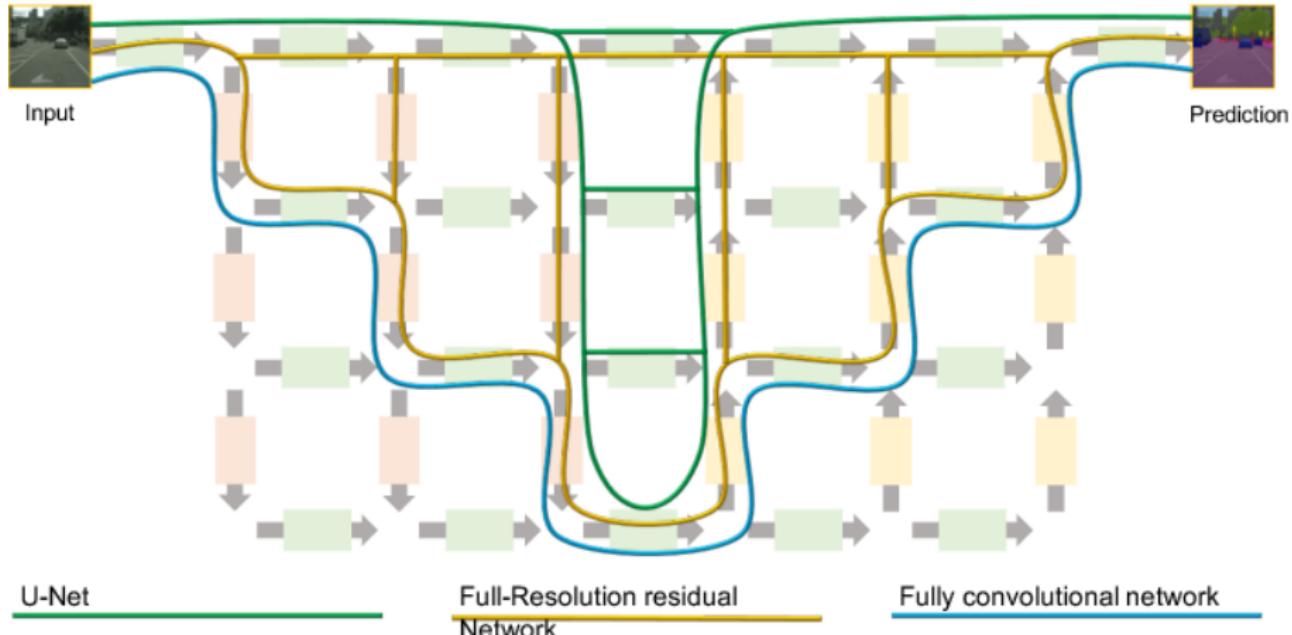


Figure 7: Architecture de Deep Learning qui sera par la suite utilisé

Exemple de segmentation par GridNet

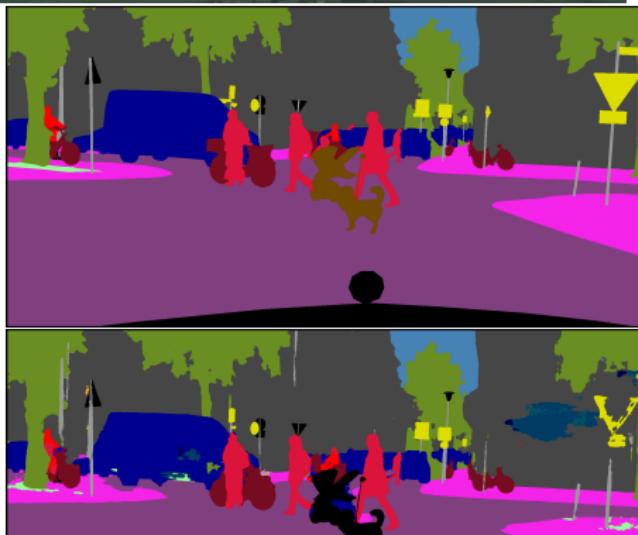


Table des matières

1 Segmentation sémantique

- Un exemple de segmentation
- Résolution du problème par le Machine Learning
- Exemple de segmentation par GridNet

2 Code en Pytorch de GridNet

3 Étude des fonctions de perte

- Fonction de coût et fonction de perte
- Minimisation direct de l'IoU
- Extension de Lovasz

Code en Pytorch de GridNet (3800)

- Réécriture de GridNet en Python.
- Structuration d'un projet complet de Machine Learning.
- Utilisation du cluster du laboratoire pour apprendre le réseau de neurone sur des GPUs.



Table des matières

1 Segmentation sémantique

- Un exemple de segmentation
- Résolution du problème par le Machine Learning
- Exemple de segmentation par GridNet

2 Code en Pytorch de GridNet

3 Étude des fonctions de perte

- Fonction de coût et fonction de perte
- Minimisation direct de l'IoU
- Extension de Lovasz

Fonction de coût et fonction de perte

Fonction de coût

- Défini par le créateur du jeu de données
- Sert de critère de performance pour la segmentation d'une image
- Pour Cityscapes Dataset c'est l'Intersection over Union qui est utilisé

$$IoU(Y, \hat{Y}) = \frac{1}{k} \sum_{k=1}^K \frac{VraiPositif_k}{FauxPositif_k + VraiNegatif_k + VraiPositif_k}$$

Fonction de perte

- Choisis par l'utilisateur du jeu de données
- Doit être dérivable pour effectuer une descente de gradient

$$crossentropy(Y, \hat{Y}) = -\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K Y_n^k \log(\hat{Y}_n^k)$$

Minimisation direct de l'IoU

Pourquoi minimiser la cross entropy...

$$-\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K Y_n^k \log(\underline{\hat{Y}_n^k})$$

alors que la performance doit être testé sur l'IoU ?

$$-\frac{1}{k} \sum_{k=1}^K \frac{\sum_{n=1}^N Y_n^k \hat{Y}_n^k}{\sum_{n=1}^N \hat{Y}_n^k + \sum_{n=1}^N Y_n^k - \sum_{n=1}^N Y_n^k \hat{Y}_n^k}$$

Passage du discret au continu

- L'IoU est discret et est défini sur $\{0, 1\}^{N \times K}$ mais les sorties du réseau de neurone sont des «probabilités» d'appartenances à chaque classe défini sur $[0, 1]^{N \times K}$

Condition nécessaire sur l'extension

$IoU(Y, \hat{Y}) = IoU_{extension}(Y, \hat{Y})$ lorsque \hat{Y} est dans l'ensemble de définition de l'IoU ($\{0, 1\}^{N \times K}$)

- Il existe une infinité de telles fonctions dérivables
- On cherche de bonne propriétés théoriques (convexité) ou de bonnes performances empiriques

Premier exemple d'extension

- Rappel de l'IoU

$$IoU(Y, \hat{Y}) = -\frac{1}{k} \sum_{k=1}^K \frac{\sum_{n=1}^N Y_n^k \hat{Y}_n^k}{\sum_{n=1}^N \hat{Y}_n^k + \sum_{n=1}^N Y_n^k - \sum_{n=1}^N Y_n^k \hat{Y}_n^k}$$

- Extension intuitive de l'IoU

$$IoU(Y, \underline{\hat{Y}}) = -\frac{1}{k} \sum_{k=1}^K \frac{\sum_{n=1}^N Y_n^k \underline{\hat{Y}}_n^k}{\sum_{n=1}^N \underline{\hat{Y}}_n^k + \sum_{n=1}^N Y_n^k - \sum_{n=1}^N Y_n^k \underline{\hat{Y}}_n^k}$$

Extension de Lovasz

- Un article [3] est paru sur exactement le même sujet que mon stage
- Utilisation de l'extension de Lovasz

$$\text{Lovasz}(Y, \underline{\hat{Y}}) = \sum_{n=0}^N (\underline{\hat{Y}}_n - \underline{\hat{Y}}_{n+1}) \text{IoU}(Y, E_n) \quad (1)$$

$$= \mathbb{E}_{\Delta}(\text{IoU}(Y, \underline{\hat{Y}} > \Delta)) \quad (2)$$



Figure 9: Représentation graphique de l'extension de Lovàsz

Autres distributions

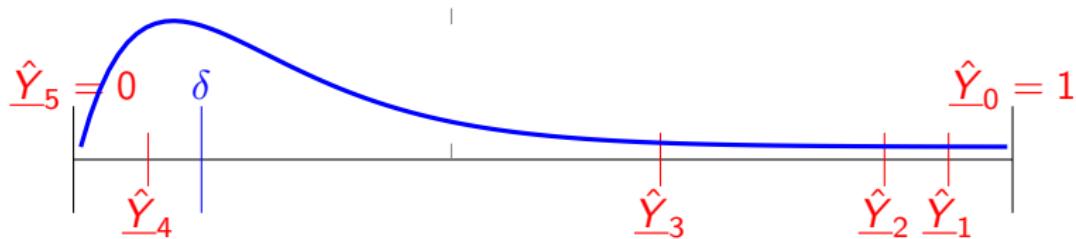


Figure 10: Représentation graphique de l'extension de Lovàsz

Modification de la distribution Δ

- Quel que soit la distribution sur $[0, 1]$, cela reste une extension de l'IoU
- Mais pour certaines distributions les performances sont parfois catastrophiques
- Des possibilités infinis de fonctions de perte qui minimisent l'IoU

Conclusion

- La segmentation d'image
- Un algorithme de Deep Learning : GridNet
- Minimisation de l'IoU
- L'extension de Lovàsz

Apport du stage

Point négatifs

- Pas de publication
- Pas d'amélioration de la performance sur la segmentation de Cityscapes Dataset

Point positifs

- Un code en Python de GridNet fonctionnel
- De nouvelles fonctions de perte utilisables
- Beaucoup de connaissances acquises : projet en Deep Learning et fonctions de pertes
- Contract de thèse au Laboratoire Hubert Curien



Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld,
Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and
Bernt Schiele.

The cityscapes dataset for semantic urban scene understanding.

In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.



Damien Fourure, Rémi Emonet, Elisa Fromont, Damien Muselet, Alain Tremeau, and Christian Wolf.

Residual conv-deconv grid network for semantic segmentation.

arXiv preprint arXiv:1707.07958, 2017.



Maxim Berman Amal Rannen Triki Matthew and B Blaschko.

The lovàsz-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks.

arXiv preprint arXiv:1705.08790, 2018.