

# Instance Segmentation for Urban Street Scenes

Francesco Bari

francesco.bari.2@studenti.unipd.it

Eleonora Signor

eleonora.signor@studenti.unipd.it

## Abstract

In questo lavoro abbiamo confrontato differenti tecniche di instance segmentation, già esistenti, sul task specifico di Urban Street Scenes. Il nostro interesse verso il topic è nato dal fatto che la segmentazione delle istanze è uno dei compiti fondamentali della visione, tuttavia si presenta ancora complesso e non del tutto esplorato.

## 1. Introduction

La segmentazione dell’immagine è il processo di separazione di questa in più segmenti, in cui ciascun pixel viene associato a un tipo di oggetto. Esistono due tipologie di segmentazione dell’immagine: la segmentazione semantica e la segmentazione d’istanza. La prima contrassegna oggetti dello stesso tipo con la medesima etichetta di classe; la seconda contrassegna oggetti dello stesso tipo e appartenenti a entità distinte con etichette di classe differenti. L’idea che abbiamo cercato di sviluppare ha riguardato il confronto di diverse metodologie di instance segmentation. La prima tecnica che abbiamo studiato è stato Mask R-CNN [1], approccio a due stadi. Questa l’abbiamo scelta alla luce del riscontro positivo che ha ricevuto dal mondo della vision research, grazie al suo framework concettualmente semplice e generale, caratterizzato da un rilevamento di oggetti d’immagine efficiente, e dalla generazione in contemporanea di una maschera di segmentazione di alta qualità per ogni istanza. La tecnica che abbiamo deciso di contrapporre a Mask R-CNN [1] è stata BlendMask [2]. Questa è invece una tecnica a uno stadio che si è presentata capace di superare le prestazioni di Mask R-CNN [1] sia a livello di previsione della maschera che per tempo di formazione, sui datasets MSCOCO 2017 [3] e LVIS [4]. Ci siamo interessati a verificare se questo rimanesse valido anche su datasets, come Cityscapes [5] e WildDash [6], appartenenti allo specifico topic di Urban Street Scenes, caratterizzati da immagini provenienti dalle strade di tutto il mondo, con molti scenari difficili. Alcuni aspetti che abbiamo testato hanno riguardato cambiamenti di backbone, profondità della rete ResNet [7] e numero di layers congelati. I risultati ottenuti ci hanno confermato quanto già annunciato dai

lavori precedenti, generalizzando BlendMask [2] come uno degli approccio a stadi più promettenti. Al termine del nostro lavoro e per non limitare la nostra analisi abbiamo fatto qualche considerazione anche su altre tecniche di instance segmentation quali SOLOv2 [8] e Deep Snake [9].

## 2. Related Work

L’approccio che abbiamo usato nel nostro lavoro è stato utilizzare come linee giuda i papers [1], [2], [8] e [9] estendendo le analisi anche su dataset di *Urban Street Scenes*.

### 2.1. Stage approach

**Mask R-CNN** [1] è una Rete Neurale Convoluzionale che si presenta all’avanguardia in termini di segmentazione dell’immagine. È la variante di una Rete Neurale Profonda che rileva gli oggetti di un’immagine e vi genera una maschera di segmentazione per ciascuna istanza. Mask R-CNN [1] è l’evoluzione successiva di Faster R-CNN [10], Rete Neurale Convoluzionale Region-based, che produce per ogni oggetto candidato 3 output: l’etichetta di classe, l’offset del riquadro di delimitazione e la maschera dell’oggetto. L’architettura della rete, Figure 1, si compone di una CNN (backbone), che processa l’immagine e estrae la feature map. Dopodichè grazie alla Region Proposal Network vengono presentate le proposte o RoI, sul quale andare a fare riconoscimento del riquadro di delimitazione e previsione delle maschera (head). Inoltre prima di generare l’output a ciascuna RoI viene applicato RoIAlign, che permette di ottenere una maschera dove il layout dell’oggetto viene mantenuto.

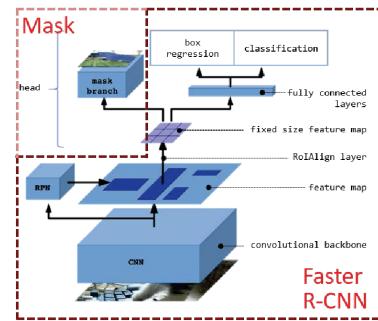


Figure 1. Mask R-CNN architecture. Source: [11], pag. 3.

**BlendMask** [2] deriva dai limiti di Mask R-CNN [1]. Gli autori del paper definiscono come Mask R-CNN [1] vincoli fortemente la velocità e la qualità di generazione delle maschere alle heads, facendo così fatica a trattare scenari complicati e ponendo un limite alla risoluzione delle maschere. Inoltre Mask R-CNN [1] si presenta come un framework poco flessibile per reti multi-task. Hanno così cercato di cobinare strategie di ricerca dall'alto verso il basso e dal basso verso l'alto in FCOS [12], one stage approach, che sembra in grado di superare le controparti a due stadi in termini di precisione. L'architettura di BlendMask [2], Figure 2, si compone di un detector network e di una mask branch. Quest'ultima è partizionata in 3 parti: il modulo inferiore che si occupa di prevedere le scores map, chiamate basi; the top layer composto da un singolo strato di convoluzione e da torri, tante quante sono le input features, con il compito di predire attention instance, e un modulo blender che unisce scores con attenzioni.

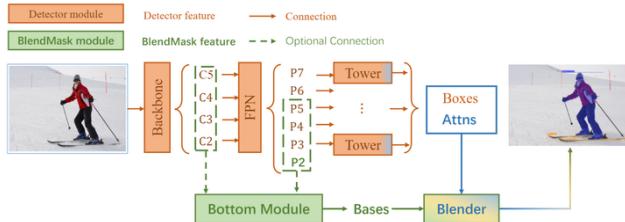


Figure 2. BlendMask architecture. Source: [2], pag. 3.

**SOLOv2** [8] è un altro approccio a singolo stage, successore di SOLO [13]. In questo caso ogni istanza di un'immagine viene segmentata dinamicamente, senza rilevamento del riquadro di delimitazione. La generazione della maschera, a differenza di Mask R-CNN [1], è disaccoppiata in mask kernel prediction e in mask feature learning. Questi due elementi sono responsabili della generazione dei convolution kernels e delle feature maps. SOLOv2 [8] riesce a ottenere risultati promettenti anche grazie all'uso della matrix non-maximum suppression (NMS) technique, che riduce le previsioni duplicate guadagnandone in minor overhead d'inferenza.

## 2.2. Contour-based approach

**Deep Snake** [9] è un approccio basato su contorni, che implementa l'idea degli algoritmi snake con learning-based approach. Deep Snake [9] si compone di una pipeline a due stadi: in un prima istanza vi è una proposta di contorno iniziale sull'oggetto di un'immagine; fatta successivamente seguire dall'uso di una Rete Neurale, che deforma iterativamente questa proposta, fino a farla combaciare esattamente con i confini propri dell'oggetto. Per l'apprendimento della struttura delle features del contorno, gli autori del paper propongono l'uso della Circular Convolution.

## 3. Datasets

I datasets di *Urban Street Scenes*, che abbiamo usato, appartengono a *Cityscapes* [5] e *WildDash* [6].

### 3.1. Cityscapes

*Cityscapes* [5] è una suite di benchmark e un dataset su larga scala per semantic urban scene understanding. È adatto per apprendere e testare metodi pixel-level e instance-level semantic labeling. Le immagini di *Cityscapes* [5] sono state create da un'insieme vasto e diversificato di sequenze video, registrate in 50 città diverse. Queste possono essere comprensive di annotazioni di alta qualità, ma anche di tipo grossolano; quest'ultime consentono di testare metodi che impiegano grandi volumi di dati debolmente etichettati. Le annotazioni contenute, fondamentali per la valutazione di un modello, sono di tipo poligonale.

In Figura3, sono indicati i gruppi, con le relative classi con cui possono venire etichettate le istanze di un'immagine.

| Group        | Classes   |
|--------------|---|
| flat         | road · sidewalk · parking · rail track*                                   |
| human        | person · rider*   |
| vehicle      | car · truck · bus · on rails · motorcycle · bicycle · caravan* · trailer* |
| construction | building · wall · fence · guard rail · bridge · tunnel*                   |
| object       | pole · pole group · traffic sign · traffic light                          |
| nature       | vegetation · terrain  |
| sky          | sky   |
| void         | ground* · dynamic* · static*  |

Figure 3. *Cityscapes* class definitions.

\* single instance annotations are available, + label not included.  
Source: <https://www.cityscapes-dataset.com/dataset-overview/>.

I datasets che abbiamo usato, appartenenti a *Cityscapes* [5], sono stati *gtFine* e *leftImg8*. Il primo è composto da annotazioni fini per 3 475 immagini di train e val, e 1 525 immagini per test set. Per non limitarci esclusivamente a lavorare con immagini annotate abbiamo deciso di utilizzare anche *leftImg8*, dataset composto da immagini di traffico urbano "standard", con train set, test set e val set; per un totale di 5 000 immagini.



Figure 4. *Cityscapes* [5] images: *gtFine* to left and *leftImg8* dataset to right.

Tuttavia i datasets di *Cityscapes* [5] anche se includono diversi mesi e stagioni, sono sempre immagini scattate in buone condizioni metereologiche. Questo aspetto ci ha

spinto ad analizzare il comportamento delle nostre tecniche di instance segmentation anche sul dataset *WildDash* [6].

### 3.2. WildDash

WildDah [6] è una suite di benchmark e un dataset per la segmentazione semantica e d'istanza per il dominio automobilistico. Le immagini, contenute nei datasets, provengono da diverse fonti da tutto il mondo. Inoltre presentano scenari, quali pioggia, oscurità, copertura stradale che sono delle vere e proprie challenge per il riconoscimento delle immagini. Questo consente di mettere in luce le carenze di una qualsiasi tecnica di instance segmentation.



Figure 5. *WildDash* [6] images: rain scenario to left and road in the desert to right.

Il dataset che abbiamo usato è *public gt package*, composto da 4 256 immagini rivolto appositamente a risolvere tasks di instance segmentation; non era tuttavia suddiviso in train, val e test set. Abbiamo di conseguenza deciso di suddividerlo manualmente, riservando 3 405 immagini come train set e 851 immagini come test set. Non abbiamo ritenuto necessario fare un'ulteriore partizione in val set, in modo da mantenere più immagini possibili nel addestramento; e assumendo che i nostri modelli, usando pesi già preaddestrati su ImageNet [14], fossero sufficientemente accurati da non richiedere model selection.

## 4. Method

Per riuscire a fare un confronto tra le diverse tecniche di instance segmentation, oggetto di questo documento, abbiamo utilizzato i seguenti approcci:

- proceduto con l'implementazione di modelli e successivamente fatto ricorso al metodo sperimentale per la valutazione;
- studiato e analizzato i risultati dei papers.

### 4.1. Dataset preparation

### 4.2. Training with fine-tuning

**Mask R-CNN** La loss utilizzata durante il training è la seguente

$$\min(L) = \min(L_{cls} + L_{box} + L_{mask})$$

$L_{cls}$  is the classification loss,  $L_{box}$  is the bounding-box loss and  $L_{mask}$  is the average binary cross-entropy loss.

**BlendMask** La loss utilizzata durante il training è la seguente

$$\min(L) = \min(\text{semantic loss}) [?]$$

### 4.3. Metrics

- AP
- numero di istanze, tempo :: accuracy visiva.confidence-threshold

### 4.4. Evaluation and inference

## 5. Experiments and results

In questa sezione descriviamo gli esperimenti che abbiamo eseguito per testare e valutare le tecniche oggetto di questo lavoro. Tali esperimenti gli abbiamo eseguiti al termine delle fasi di studio e codifica.

### 5.1. Backbone

La seconda serie di esperimenti, che abbiamo compiuto, riguarda la definizione della backbone. Tutte le tecniche a stadi, oggetto del confronto, sono dotate del suddetto modulo inferiore, per cui ci è risultato semplice uniformare le scelte architettoniche in modo da poter compiere una valutazione oggettiva. Le tecniche che abbiamo confrontato sono state Mask R-CNN e BlendMask.

Le configurazioni costanti delle reti sono image size ..., numero massimo di iterazioni, learning rate ..., step size a ... e fine-tuning esclusivamente agli ultimi 2 livelli.

| Method and architecture                              | Cityscapes AP | WildDash AP |
|--|---------------|-------------|
| Mask R-CNN + ResNet50<br>+ C4 + Base-RCNN-C4         |               |             |
| Mask R-CNN + ResNet50<br>+ DC5 + Base-RCNN-DilatedC5 |               |             |
| Mask R-CNN + ResNet50<br>+ FPN + Base-RCNN-FPN       |               |             |

Table 1. Backbone Mask R-CNN result.

Per BlendMask, oltre a settare le configurazioni costanti, avvalendoci dei risultati presentati in ... abbiamo settato  $R = 56$ ,  $M = 14$ ,  $K = 4$ , sampling method for bottom bases bilinear pooling, interpolation method for top-level attentions bilinear upsampling and semantic loss. Inoltre abbiamo deciso di testare vari tipi di decoder: ProtoNet and DeepLabv3+.

| Method and architecture  | <i>Cityscapes AP</i> | <i>WildDash AP</i> |
|--|----------------------|--------------------|
| BlendMask with decoder ProtoNet<br>+ ResNet50 + FPN + Base-550                               |                      |                    |
| BlendMask with decoder ProtoNet<br>+ ResNet50 + deformable convolution<br>+ FPN + Base-550   |                      |                    |
| BlendMask with decoder DeepLabv3+<br>+ ResNet50 + FPN + Base-550                             |                      |                    |
| BlendMask with decoder DeepLabv3+<br>+ ResNet50 + deformable convolution<br>+ FPN + Base-550 |                      |                    |

Table 2. Backbone BlendMask result.

## 5.2. Deepness

Una terza serie di esperimenti ha riguardato lo studio della profondità delle reti ResNet.

I parametri di configurazione non definiti in modo esplicito, sono le medesime di quelle riportate nella sezione §5.1.

| Method and architecture   | <i>Cityscapes AP</i> | <i>WildDash AP</i> |
|---|----------------------|--------------------|
| BlendMask with decoder ProtoNet<br>+ ResNet101 + FPN + Base-BlendMask                             |                      |                    |
| BlendMask with decoder ProtoNet<br>+ ResNet101 + deformable convolution<br>+ FPN + Base-BlendMask |                      |                    |

Table 3. Deepness BlendMask result.

## 5.3. Freeze levels

Per la quarta serie di esperimenti ci siamo voluti concentrare sul numero di layers da "scongelare" di ResNet durante il re-training dei pesi.

I parametri di configurazione non definiti in modo esplicito, sono le medesime di quelle riportate nella sezione §5.1.

| Method and architecture  | <i>Cityscapes AP</i> | <i>WildDash AP</i> |
|--|----------------------|--------------------|
| Mask R-CNN + ResNet101 + FPN<br>1 layers freeze  |                      |                    |
| Mask R-CNN + ResNet101 + FPN<br>3 layers freeze  |                      |                    |
| BlendMask with decoder ProtoNet<br>+ ResNet101 + FPN + Base-BlendMask<br>1 layers freeze |                      |                    |
| BlendMask with decoder ProtoNet<br>+ ResNet101 + FPN + Base-BlendMask<br>3 layers freeze |                      |                    |

Table 4. Freeze layers result.

## 5.4. Own best models

Come ultima serie di esperimenti abbiamo cercato di individuare i modelli migliori, per ciascuna le due tecniche di instance segmentation in esame in questa sezione; tenendo conto della possibilità di allenare ciascun modello solo su una singolo macchina e 1 GPU.

I parametri di configurazione non definiti in modo esplicito, sono le medesime di quelle riportate nella sezione §5.1.

| Dataset | method and architecture | AP |
|---------|-------------------------|----|
|         |                         |    |

Table 5. Own best models result.

## 6. Conclusion

Dai nostri esperimenti siamo in grado di dire che le tecniche a uno stadio e box based, opportunamente modificate, funzionano meglio rispetto a metodi a due stadi (di cui re è Mask R-CNN [1]), sia in termini di AP precision che di tempo GPU. Tali effetti positivi sono probabilmente causati dall'ibridazione di metodi top down e bottom up, e dall'rilevamento degli oggetti senza ancoraggio (come accade per BlendMask [2]). In aggiunta, recenti studi riportati in [8] e in parte mostrati in Figure 6, provano che uno stage approach box-free combinato a matrix NMS, come lo è SOLOv2 [8], si dimostra molto competitivo nei confronti di BlendMask [2].

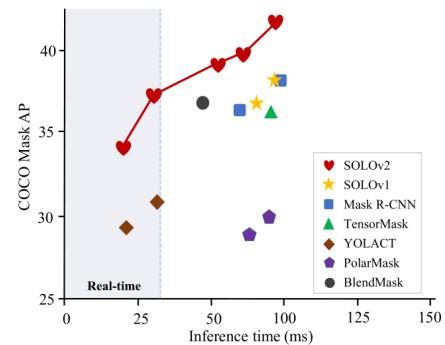


Figure 6. Comparison between SOLOv2 and other stage approaches. Source: [8], pag. 2.

Inoltre gli stage approach non sono gli unici metodi possibili per risolvere tasks di instance segmentation. Per esempio esiste Deep Snake [9], counter based approach che supera Mask R-CNN [1] sia in termini di velocità di inferenza che AP precision, come mostrato dalla nella tabella in Figure 7, e ha il potenziale per essere un buon competitor di SOLOv2 [8]. Una possibile estensione futura, in questa direzione, può essere utilizzare l'approccio basato su contorni per il rilevamento dei riquadri di delimitazione in BlendMask [2].

|                  | training data | fps | AP [val] | AP   | AP <sub>50</sub> |
|------------------|---------------|-----|----------|------|------------------|
| SGN [26]         | fine + coarse | 0.6 | 29.2     | 25.0 | 44.9             |
| PolygonRNN++ [1] | fine          | -   | -        | 25.5 | 45.5             |
| Mask R-CNN [18]  | fine          | 2.2 | 31.5     | 26.2 | 49.9             |
| GMIS [28]        | fine + coarse | -   | -        | 27.6 | 49.6             |
| Spatial [31]     | fine          | 11  | -        | 27.6 | 50.9             |
| PANet [27]       | fine          | <1  | 36.5     | 31.8 | 57.1             |
| Deep snake       | fine          | 4.6 | 37.4     | 31.7 | 58.4             |

Figure 7. Results on *Cityscapes* val (AP [val] column) and test (remaining columns) sets. Source: [9], pag. 7.

## References

- [1] Kaiming He and Georgia Gkioxari and Piotr Dollár and Ross Girshick. Mask R-CNN. CoRR, 2018.

- [2] Hao Chen, Kunyang Sun, Zhi Tian, Chunhua Shen, Yongming Huang and Youliang Yan. BlendMask: Top-Down Meets Bottom-Up for Instance Segmentation. CoRR, 2020.
- [3] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. CoRR, 2014.
- [4] Agrim Gupta, Piotr Dollár and Ross B. Girshick. LVIS: A Dataset for Large Vocabulary Instance Segmentation. CoRR, 2019.
- [5] 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. CoRR, 2016.
- [6] Zendel, Oliver and Honauer, Katrin and Murschitz, Markus and Steininger, Daniel and Dominguez, Gustavo Fernandez. WildDash - Creating Hazard-Aware Benchmarks. Proceedings of the European Conference on Computer Vision, (ECCV), 2018.
- [7] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. Deep Residual Learning for Image Recognition. CoRR, 2015.
- [8] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li and Chunhua Shen. SOLOv2: Dynamic, Faster and Stronger. CoRR, 2020.
- [9] Sida Peng, Wen Jiang, Huaijin Pi, Hujun Bao and Xiaowei Zhou. Deep Snake for Real-Time Instance Segmentation. CoRR, 2020.
- [10] Shaoqing Ren, Kaiming He, Ross B. Girshick, Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. CoRR, 2015.
- [11] Bienias, Lukasz & n, Juanjo & Nielsen, Line & Alstrøm, Tommy. Insights Into The Behaviour Of Multi-Task Deep Neural Networks For Medical Image Segmentation. 2019.
- [12] Zhi Tian, Chunhua Shen, Hao Chen and Tong He. FCOS: Fully Convolutional One-Stage Object Detection. CoRR, 2019.
- [13] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang and Lei Li. SOLO: Segmenting Objects by Locations. CoRR, 2019.
- [14] J. Deng, W. Dong, R. Socher, L. -J. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.
- [15] Xu, Jingyi and Zhang, Zilu and Friedman, Tal and Liang, Yitao and Van den Broeck, Guy. A Semantic Loss Function for Deep Learning with Symbolic Knowledge. Proceedings of the 35th International Conference on Machine Learning, 2018.