Experience feedback using Representation Learning for Few-Shot Object Detection on Aerial Images

Pierre Le Jeune
L2TI (UR 3043), USPN[1] & COSE

Anissa Mokraoui
L2TI (UR 3043), USPN[1]

Mustapha Lebbah
LIPN (UR 7030), USPN[1]

Hanene Azzag
LIPN (UR 7030) USPN[1]

20th IEEE International Conference for Machine Learning and Applications
December 13, 2021

[1] *Université Sorbonne Paris Nord*

Overview of the presentation

1. Definition of Few-Shot Object Detection
2. Related work
3. Proposed Prototypical Faster R-CNN
4. Experimental Results
5. Analysis and Conclusion

1. Few-Shot Object Detection

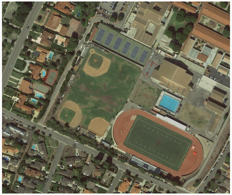
n-way k-shot object detection

Given support examples $\{(\boldsymbol{x}_1, \boldsymbol{a}_1), \dots, (\boldsymbol{x}_{nk}, \boldsymbol{a}_{nk})\}$ it consists in detecting all occurrences of classes in $\mathcal{C}$ ($|\mathcal{C}| = \boldsymbol{n}$) in a query image $\boldsymbol{x}_{\boldsymbol{q}}$.

1. Few-Shot Object Detection

n-way k-shot object detection

Given support examples $\{(\boldsymbol{x}_1, \boldsymbol{a}_1), \ldots, (\boldsymbol{x}_{nk}, \boldsymbol{a}_{nk})\}$ it consists in detecting all occurrences of classes in $\mathcal{C}$ ($|\mathcal{C}| = \boldsymbol{n}$) in a query image $\boldsymbol{x}_{\boldsymbol{q}}$.

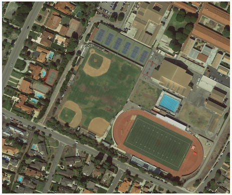


Query image

1. Few-Shot Object Detection

***n*-way *k*-shot object detection**

Given support examples $\{(\boldsymbol{x}_1, \boldsymbol{a}_1), \ldots, (\boldsymbol{x}_{nk}, \boldsymbol{a}_{nk})\}$ it consists in detecting all occurrences of classes in $\mathcal{C}$ ($|\mathcal{C}| = \boldsymbol{n}$) in a query image $\boldsymbol{x_q}$.

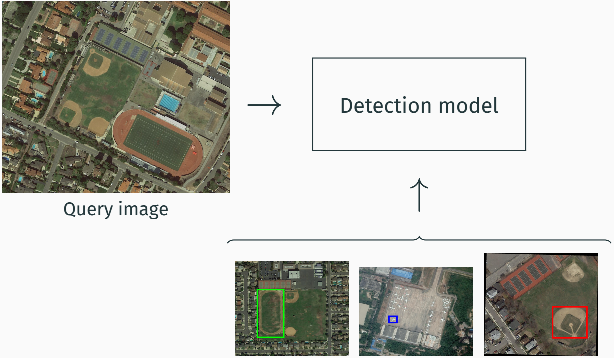


Query image

Support examples

***n*-way *k*-shot object detection**

Given support examples $\{(\mathbf{x}_1, \mathbf{a}_1), \ldots, (\mathbf{x}_{nk}, \mathbf{a}_{nk})\}$ it consists in detecting all occurrences of classes in $\mathcal{C}$ ($|\mathcal{C}| = \mathbf{n}$) in a query image $\mathbf{x}_q$.

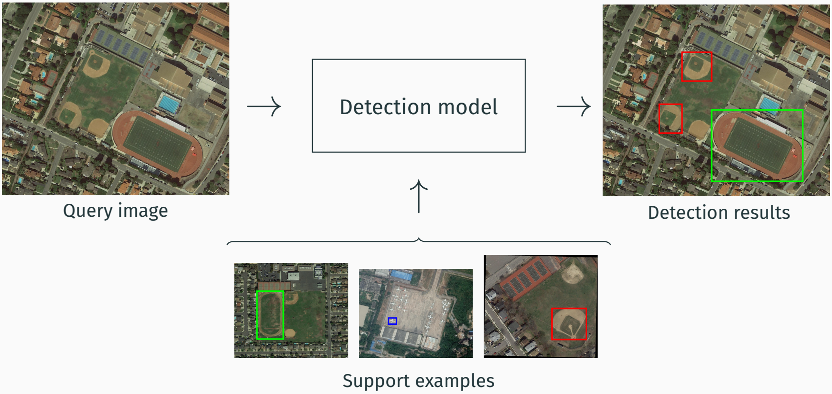


Support examples

1. Few-Shot Object Detection

n-way k-shot object detection

Given support examples $\{(x_1, a_1), \ldots, (x_{nk}, a_{nk})\}$ it consists in detecting all occurrences of classes in $\mathcal{C}$ ($|\mathcal{C}| = n$) in a query image x_q.



2.1 Related work - Faster R-CNN for Object Detection

Faster R-CNN (Ren et al. 2015) is a 2-stages approach for Object Detection

- Backbone network: large CNN to extract features.
- Region Proposal Network (RPN): lightweight CNN that proposes boxes.
- Classification and regression head: MLP that predicts box coordinates and class.

Robust and well-performing architecture, extensively tested in literature.

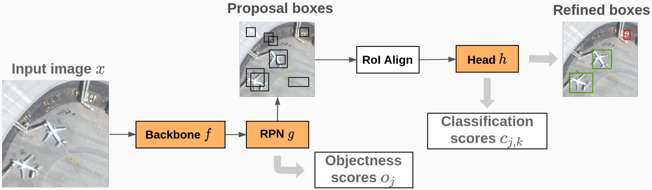


Figure 1: Faster R-CNN architecture introduced by (Ren et al. 2015)

Prototypical networks (Snell, Swersky, and Zemel 2017) have been introduced for Few-Shot Classification

- Learn an embedding function
- Compute prototypes vectors from available class examples
- Classify an image according to the distance between its representation and the prototypes

The embedding space is semantically organized: easy adaptation to new classes.

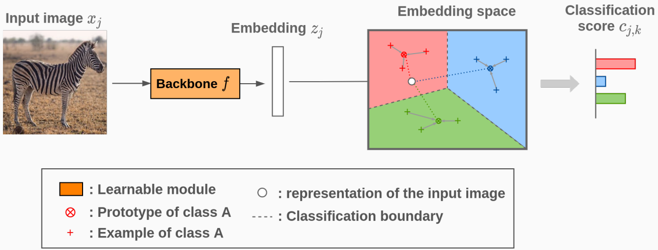


Figure 2: Diagram explaining the principle of Prototypical Networks (Snell, Swersky, and Zemel 2017)

3.1 Prototypical Faster R-CNN - Architecture

Main principle: integrate prototypical networks inside Faster R-CNN.

3.1 Prototypical Faster R-CNN - Architecture

Main principle: integrate prototypical networks inside Faster R-CNN.

RPN: multi-class prototypes but only outputs objectness score (i.e. binary classification).

$$o_j = \max_{c \in C_i} \exp\left(\frac{-d(z_j, p_c)^2}{2\sigma^2}\right)$$

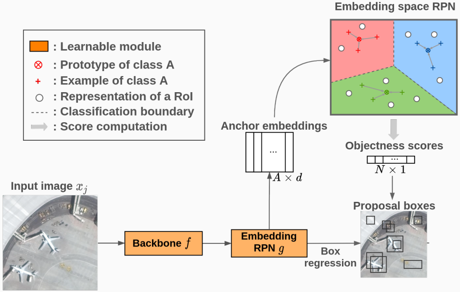


Figure 3: Prototypical Faster-RCNN architecture.

Main principle: integrate prototypical networks inside Faster R-CNN.

RPN: multi-class prototypes but only outputs objectness score (i.e. binary classification).

$$o_j = \max_{c \in C_i} \exp\left(\frac{-d(z_j, p_c)^2}{2\sigma^2}\right)$$

Classification head: prototypical networks attribute class scores to RoI extracted from RPN boxes (Karlinsky et al. 2019).

$$p(c|x_{j,a}) = \frac{\exp\left(\frac{-d(z_j, p_c)^2}{2\sigma^2}\right)}{\sum\limits_{c' \in C_i \cup \{\varnothing\}} \exp\left(\frac{-d(z_j, p_{c'})^2}{2\sigma^2}\right)}$$

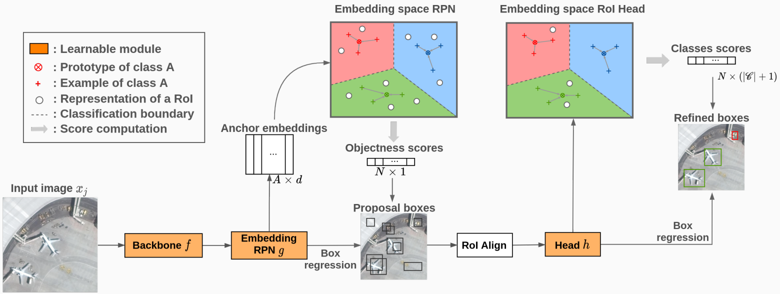


Figure 3: Prototypical Faster-RCNN architecture.

3.2 Prototypical Faster R-CNN - Training strategy

Class separation: classes are split into base classes $\mathcal{C}_{base}$ and novel classes $\mathcal{C}_{novel}$ before training.

Training is done episodically.

Algorithm 1 : Training procedure

1: **for** i in range $[1, N_{ep}]$ **do**
2: Randomly sample $\mathcal{C}_{ep} \subset \mathcal{C}_{base}$
3: Build a support set with k examples for each $c \in \mathcal{C}_{ep}$ from the dataset
4: Compute prototypes from the support set
5: Sample a query set Q_{ep} containing all classes from $\mathcal{C}_{ep}$ from the dataset
6: Optimize the objective with Q_{ep}
7: **end for**

3.3 Prototypical Faster R-CNN - Loss functions

Region Proposal Network

$$\mathcal{L}^{R}_{reg}(b^{R}_{i}, \hat{b}^{R}_{i}) = \mathrm{SmoothL1Loss}(b^{R}_{i}, \hat{b}^{R}_{i}),$$

$$\mathcal{L}^{R}_{obj}(o_i, \hat{o}_i) = \hat{o}_i \log(o_i) + (1 - \hat{o}_i)\log(1 - o_i),$$

b^{H}_{i} box prediction from the RPN
o_i objectness score from the RPN

Classification and regression head

$$\mathcal{L}^{H}_{reg}(b^{H}_{j}, \hat{b}^{H}_{j}) = \mathrm{SmoothL1Loss}(b^{H}_{j}, \hat{b}^{H}_{j}),$$

$$\mathcal{L}^{H}_{cls}(c_j, \hat{c}_j) = -\log(c_j).$$

b^{H}_{j} box prediction from the head
c_j classification scores from the head

The overall objective is defined as:

$$\mathcal{L} = \mathcal{L}^{R}_{reg} + \mathcal{L}^{R}_{obj} + \mathcal{L}^{H}_{reg} + \mathcal{L}^{H}_{cls}.$$

4.1 Experimental Results - mAP on DOTA

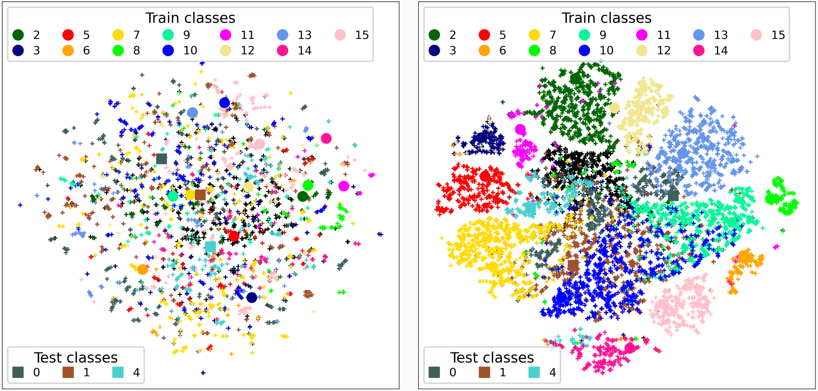
Experimental Protocol: Training with base classes and evaluation on novel classes.

- DOTA dataset (Xia et al. 2018): aerial images (*16 classes, 200k objects*)
- 2 distinct class splits
- Episodic evaluation with random support set
- No fine-tuning

	# Shots	1	3	5	10
Split A	Base classes	0.275 ± 0.01	0.352 ± 0.02	0.390 ± 0.01	0.384 ± 0.02
	Novel classes	0.047 ± 0.02	0.024 ± 0.01	0.038 ± 0.01	0.041 ± 0.01
Split B	Base classes	0.415 ± 0.03	0.392 ± 0.03	0.434 ± 0.02	0.414 ± 0.03
	Novel classes	0.08 ± 0.01	0.101 ± 0.02	0.121 ± 0.01	0.101 ± 0.02

Table 1: Mean average precision over 5 runs on DOTA dataset with 95% confidence interval. Results are given for two different base/novel classes split. Split A: [plane, ship, and tennis court], Split B: [harbor, helicopter, and soccer ball field] (only test classes are given).

4.2 Experimental Results - Embedding space visualization



(a) Before training

(b) After training

Figure 4: TSNE visualization on the embedding space, before and after training. Training organizes this space semantically and reduces the threadlike patterns representing close patches in the input image.

Key difficulties and potential solutions:

5. Analysis and conclusion

Key difficulties and potential solutions:

- Faster R-CNN is not well suited for few-shot object detection
  - Low performance in RPN can cripple classification head.
  - Two-stages and anchors approaches bring unnecessary complexity.

5. Analysis and conclusion

Key difficulties and potential solutions:

- Faster R-CNN is not well suited for few-shot object detection
  - Low performance in RPN can cripple classification head.
  - Two-stages and anchors approaches bring unnecessary complexity.

  → **Change detection framework for one-stage w/o anchors, e.g. FCOS (Tian et al. 2019).**

5. Analysis and conclusion

Key difficulties and potential solutions:

- Faster R-CNN is not well suited for few-shot object detection
  - Low performance in RPN can cripple classification head.
  - Two-stages and anchors approaches bring unnecessary complexity.

  $\rightarrow$ **Change detection framework for one-stage w/o anchors, e.g. FCOS (Tian et al. 2019).**

- Support examples are unlikely optimal for a query image
  - Semantic information may be dominated by background within a patch.
  - Classification scores depend on background similarity between examples and the patch.

5. Analysis and conclusion

Key difficulties and potential solutions:

- Faster R-CNN is not well suited for few-shot object detection
  - Low performance in RPN can cripple classification head.
  - Two-stages and anchors approaches bring unnecessary complexity.

  $\rightarrow$ **Change detection framework for one-stage w/o anchors, e.g. FCOS (Tian et al. 2019).**

- Support examples are unlikely optimal for a query image
  - Semantic information may be dominated by background within a patch.
  - Classification scores depend on background similarity between examples and the patch.

  $\rightarrow$ **Adapt prototypes to match query through an attention mechanism.**

5. Analysis and conclusion

Key difficulties and potential solutions:

- Faster R-CNN is not well suited for few-shot object detection
  - Low performance in RPN can cripple classification head.
  - Two-stages and anchors approaches bring unnecessary complexity.

  $\rightarrow$ **Change detection framework for one-stage w/o anchors, e.g. FCOS (Tian et al. 2019).**

- Support examples are unlikely optimal for a query image
  - Semantic information may be dominated by background within a patch.
  - Classification scores depend on background similarity between examples and the patch.

  $\rightarrow$ **Adapt prototypes to match query through an attention mechanism.**

Thank you for your attention

Any questions ?

pierre.lejeune@edu.univ-paris13.fr

https://pierlj.github.io

References i

Karlinsky, Leonid et al. (2019). "Repmet: Representative-based metric learning for classification and few-shot object detection". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5197–5206.

Ren, Shaoqing et al. (2015). "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *Advances in neural information processing systems* 28, pp. 91–99.

Snell, Jake, Kevin Swersky, and Richard Zemel (2017). "Prototypical Networks for Few-shot Learning". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc.

Tian, Zhi et al. (2019). "Fcos: Fully convolutional one-stage object detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9627–9636.

Xia, Gui-Song et al. (2018). "DOTA: A large-scale dataset for object detection in aerial images". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3974–3983.