Cascaded Zoom-in Detector for High Resolution Aerial Images - Supplementary Material

Akhil Meethal Eric Granger Marco Pedersoli
LIVIA lab, Dept. of Systems Engineering, ÉTS Montreal, Canada
akhilpm135@gmail.com, {marco.pedersoli, eric.granger}@etsmtl.ca

1. Appendix

This appendix provides additional ablation studies and analyses using the VisDrone dataset.

1.1. Impact of hyperparameters in the crop labeling algorithm

In this section, we analyzed the impact of different hyperparameters in the crop labeling algorithm. Subsequently, we observed that they can be easily tuned.

1.1.1 Impact of N in the crop-labeling algorithm

When we performed iterative merging, we set the number of merging steps N=2. In this section, we empirically verify the impact of other possible N values. Table 1 shows the comparison. When N=1, we have too many small noisy crops. The optimal quality crops are obtained when N = 2. When N = 3, the number of crops reduces, and crops tend to grow significantly to cover many background regions. Thus we see a reduction in performance. A visualization of the crops shows further evidence regarding the quality of crops when iterative merging is used in the crop-labeling. Figure 1 shows the comparison when crop-labeling is performed with different iteration values N. When N = 1, regardless of whether we scale the boxes by pixel or a scaling factor, we get too many small crops often containing a few objects, violating our quality constraints. It is also producing crops around large objects in the image. N = 2 gives the best quality crops, optimal in number according to the object density, and encloses mostly the small objects. N = 3 enlarges the crops so much covering background regions, also very large crops are getting filtered out (fig 1 first row, third column). This establishes that our crop labeling algorithm is respecting the specified quality constraints.

1.1.2 Impact of the θ on the Crop Labeling Algorithm.

In Table 2, we studied the impact of the θ parameter in the crop labeling algorithm. From the results, it can be observed that a small value of the θ is preferred. The bounding boxes of the small objects exhibit low IoU values, hence maximum connections are observed for low values of θ. A large value for the overlap parameter θ affects the performance significantly. This is expected, as we increase the threshold, the connections in the crop labeling algorithm reduce, and hence the number of density crops discovered also reduces. This will subsequently move towards the baseline case where no density crops are used both at the train and test time.

N	# crops	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
1	62677	31.26	54.55	31.50	23.83	40.78	50.07
2	14018	33.02	57.87	33.09	25.74	42.93	41.44
3	2227	31.14	55.22	30.88	23.99	40.26	40.43

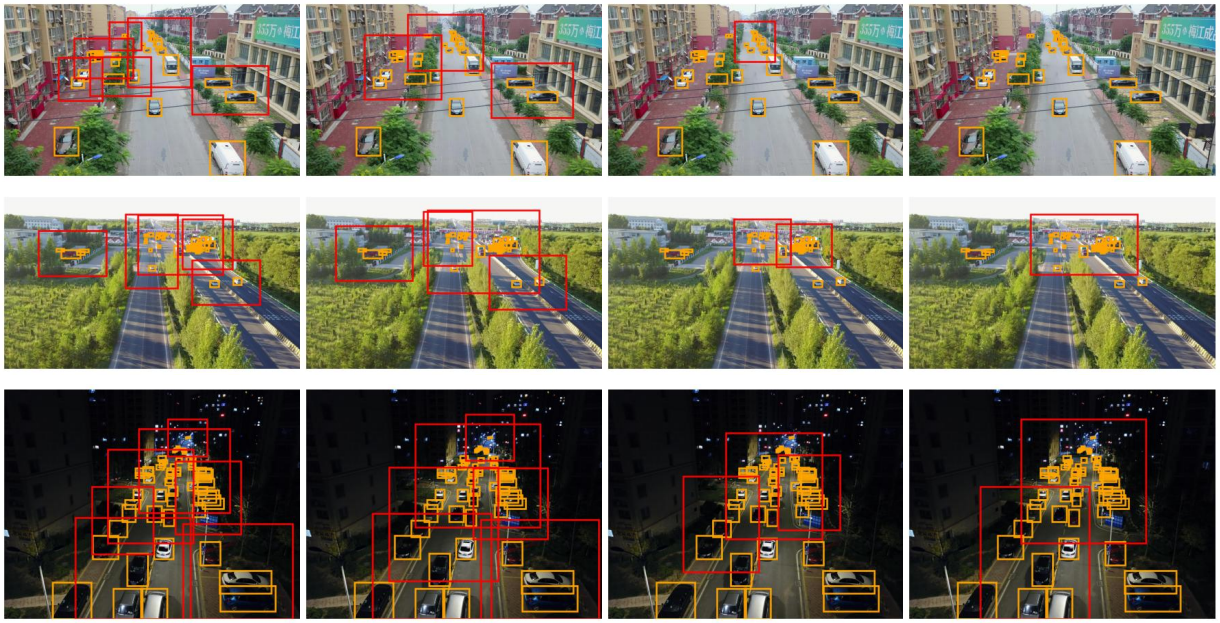
Table 1. Impact on performance of the number of iterative merging steps N used the density crop labeling algorithm (results in %).

θ Value	# crops	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
0.1	14018	33.02	57.87	33.09	25.74	42.93	41.44
0.2	12652	32.80	57.62	32.82	25.50	42.43	43.56
0.3	10222	32.66	57.25	32.70	25.57	42.33	44.10
0.4	7551	32.08	56.69	31.84	24.26	42.47	43.99
0.5	4836	31.48	55.87	31.13	24.23	41.10	38.19

Table 2. Impact on performance of the overlap threshold θ in the density crop labeling algorithm (results in %).

1.1.3 Impact of π in the crop-labeling algorithm

We used a maximum size limit for the crops to ignore oversized crops. The size parameter π in the crop-labeling algorithm represents the ratio of the area of the crop to that of the image and allows us to ignore those crops whose area ratio is above a threshold. This parameter serves only to filter out unusually big crops spanning a big portion of the image and requires minimal tuning. Table 3 shows the results in the case of the VisDrone dataset. As we can see, there are few crops growing significantly when using a 2-stage iterative merging. So setting π = 0.3 provides the best quality crops



(a) 1-stage pixel (b) 1-stage factor (c) 2-stage (d) 3-stage

Figure 1. Comparison of different crop-labeling algorithms. (a) 1-stage with scaling by pixel (b) 1-stage with scaling by a scaling factor. (c) our 2-stage iterative merging. (d) 3-stage iterative merging. 1-stage merging is producing too many crops even grouping larger objects in the image; some crops contain very few objects. 2-stage merging produces the optimal number of crops by mostly enclosing small objects in the image. 3-stage merging is producing very few crops, often spanning to large areas in the background regions. Sometimes the crops are even disappearing due to the size constraint limiting the crop size (eg: col 4 in the first row).

π	# crops	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
0.1	11574	32.23	56.27	32.32	24.53	42.49	41.64
0.2	13732	32.74	57.44	32.56	25.64	42.39	41.26
0.3	14018	33.02	57.87	33.09	25.74	42.93	41.44
0.4	14018	33.02	57.87	33.09	25.74	42.93	41.44

Table 3. Impact on performance of the maximum crop size π used the density crop labeling algorithm (results in %).

on average.

From these experiments on the hyperparameters of the crop labeling algorithm (π, θ, N), we can observe one thing in common; it is easy to tune their values. N takes discrete values between(1, 3]. θ should be ideally small (< 0.2) to have connections between scaled boxes. It is not sensible to use crops above 50% of the size of the image. The maximum crop size should be at least 10% of the image, else we won't get many crops. So π should be something in between [0.1, 0.5).

1.2. Impact of different backbones

Table 4 compares the result of our approach with ResNet-101 and ResNet-50 [2] backbones. The trend is similar. For ResNet-101 also, the result without high-resolution features P2 is close to that of with P2, thus we

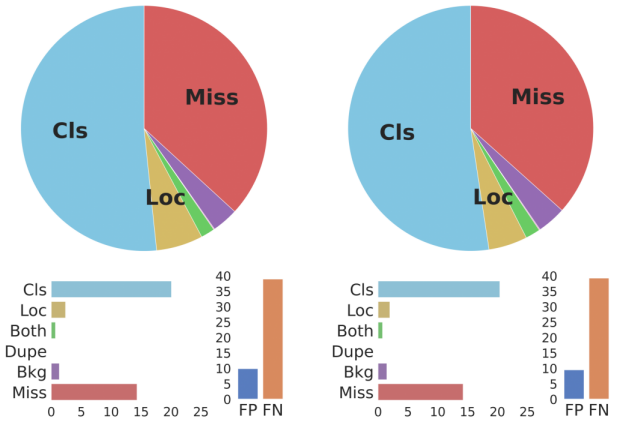
Settings	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l	FPS
Without P2							
Baseline R-50	29.48	51.68	29.55	22.33	38.66	39.30	26.31
Baseline R-101	30.98	54.74	30.55	22.84	41.06	42.30	23.79
CZ Det. R-50	33.02	57.87	33.09	25.74	42.93	41.44	11.64
CZ Det. R-101	33.91	59.07	33.87	26.14	44.64	45.03	10.01
With P2							
Baseline R-50	30.81	55.06	30.68	23.97	39.19	41.17	18.25
Baseline R-101	31.41	55.47	31.29	23.97	40.56	43.83	16.48
CZ Det. R-50	33.22	58.30	33.16	26.06	42.58	43.36	8.44
CZ Det. R-101	34.36	59.65	34.55	26.96	44.23	42.14	6.18

Table 4. The performance of Baseline and CZ Detectors with the R-50 and R-101 backbones using Faster RCNN [3] (results in %).

are getting a significant boost even when high-resolution features are not used. This illustrates the advantages of using density crops over sparse convolutions on high resolution features for small object detection as proposed in [4]. Figure 2 shows additional detection results on both VisDrone and DOTA datasets along with the high-quality density crops predicted by the detector.

To understand whether our approach is creating any drastic changes in the learning dynamics of the baseline detector, we analyzed the error distribution of the detectors trained with our approach and the baseline training using TIDE [1]. Figure 3 shows the results. It is evident that

Figure 2. Additional detection results: The first row shows detection on the DOTA images, the second row shows detection on the VisDrone images. Red boxes in each image shows the confident crops detected.



(a) Baseline (b) With density crops

Figure 3. Error analysis: Baseline vs With density crops. Error types are: **Cls**: localized correctly but classified incorrectly, **Loc**: classified correctly but localized incorrectly, **Both**: both cls and loc error, **Dupe**: duplicate detection error, **Bkg**: detected background as foreground, **Miss**: missed ground truth error.

learning a new class "density crop" with augmented crops and two-stage inference is not introducing any significant change in the behavior of the detector. The only change we observed is the reduction of the localization error, thus not altering any other aspects of the detector. Thus our method keeps the detector intact and only improves the performance of small object detection leveraging density crops.

References

[1] D. Bolya, S. Foley, J. Hays, and J. Hoffman. TIDE: A general toolbox for identifying object detection errors. In *ECCV*, 2020. 2

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. 2016. 2

[3] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 2

[4] C. Yang, Z. Huang, and N. Wang. Querydet: Cascaded sparse query for accelerating high-resolution small object detection. In *CVPR*, 2022. 2