Galaxy Counting with CounTR on a new Dataset for Object Counting

Arturo Ghinassi and Valentino Sacco

January 11, 2023

Abstract

The counting problem has recently become more challenging after the shift from detection based architectures to pure counting models. The state of the art at the time of the beginning of this work is CounTR: a transformer-based architecture for generalised visual object counting that exploits self similarity of to-be-counted entities. This architecture has proven to be quite performing on generic objects without being trained on those specific classes, but is it going to perform well when we shift to FITS images? In this work we prove that this is possible, given the right adjustments.

1 Introduction and Related Work

In the original paper[1] the model has been tested on the FSC147 (multi-class and general-purpose dataset) and CARPK dataset (drone images of parking lots) with impressive results; in this project we tried to stress its counting ability when applied to scenarios different from the original dataset. In the specific, we used their model, pretrained on FSC147 to reconstruct missing parts of an image, on a newly created dataset that we called **Galaxy Counting Dataset (GCD)** (single-class dataset).

2 Related Work

For a deeper understanding of the model we leave the reader to the paper *Transformer-based Generalised Visual Counting* [1]. Regarding the task of detecting galaxies we highlight *Galaxy detection and identification using deep learning and data augmentation* [2] where there is a good overview of how data augmentation helps the task of detecting galaxies.

3 Methods

CounTR's proposed method of object counting consists of estimating a density map over the original image whose sum returns the estimated number of objects in the image (we leave you to the paper for a better understanding). The models are tested in both the few and zero-shot counting cases, where as input are respectively given a few example patches or none. For the few-shot case feeding examples of different colors and shapes helps with the invariance, whereas in the zero-shot the model is left to learn alone. Starting from pretrained weights, we fine-tuned the model on the new datasets, quite different from those in the original paper[1], implementing data augmentation pipelines to help study the model's counting ability on objects of much smaller size than the patches (16x16) of the ViT Encoder. Lastly, testing was run on both CounTR initial code aswell as our implementation, to better fit the data in use.

3.1 Training

For the sake of finetuning our datasets have been split for train/val/test in 80/10/10 percent sizes. For the training pipeline we undertook different approaches changing most of the steps as: removing blending and mosaicing (as they seemed pointless for this case study) and resulting with random steps such as noise, color jittering, and flipping.

In both cases (ours and the original paper[1] implementation) the model was trained for 100 epochs and Mean Absolute Error and Root Mean Squared Error were used as metrics.

The model was trained mixing zero and few-shot scenarios, trying different combination of hyperparameters to refine the learning process. The resulting top configuration consists of the original paper[1] loss and Cosine Annealing as learning scheduler.

4 Dataset

the Galaxy dataset was generated from a 25-megapixel FITS image and the corresponding ground truth mask of individual galaxies. The image was divided into patches of (384, 384) pixels (fig. 1) for which density maps were extracted from the segmentation mask. The peculiarity of the dataset, in addition to having a single class, is the presence of strong

background noise due to other celestial bodies present in the image and difficult to distinguish from the galaxies that could mislead the model.

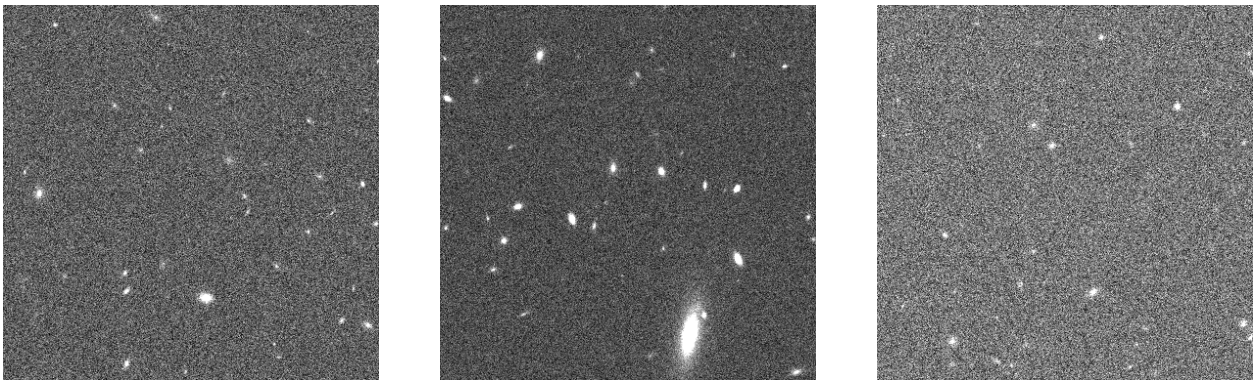

Figure 1: Sample Images of GCD

5 Results

In the case of Galaxy, finetuning with the first and original pipeline completely overfit the data. Having no validation set disallowed us to know when to stop training and left us with a training MAE and RMSE of respectively 3.08 and 3.85 which in testing became 314.39 and 366.23 (fig. 3). We noticed an upscaling technique in the original testing code, which crops patches of the images and upscales them to perform counting. Since stars and galaxies are quite similar in FITS this introduced a lot of noise and confused the model in miscounting stars as galaxies, even though it was still very sure of the real results.
The second pipeline returned very surprising results: the best model (epoch 91) has a MAE of 2.60 in train and 2.58 in validation, which then becomes 2.08 in test. The awesome results can be seen in fig. 2

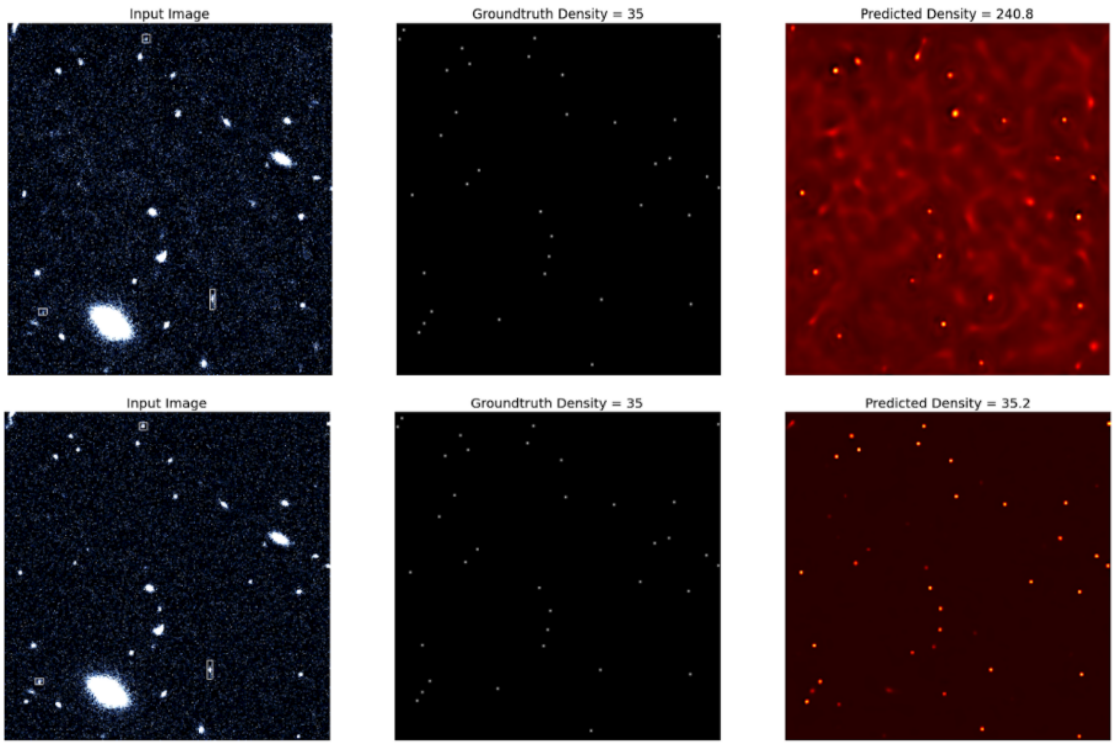

Figure 2: Paper[1] vs Our Implementation Predictions

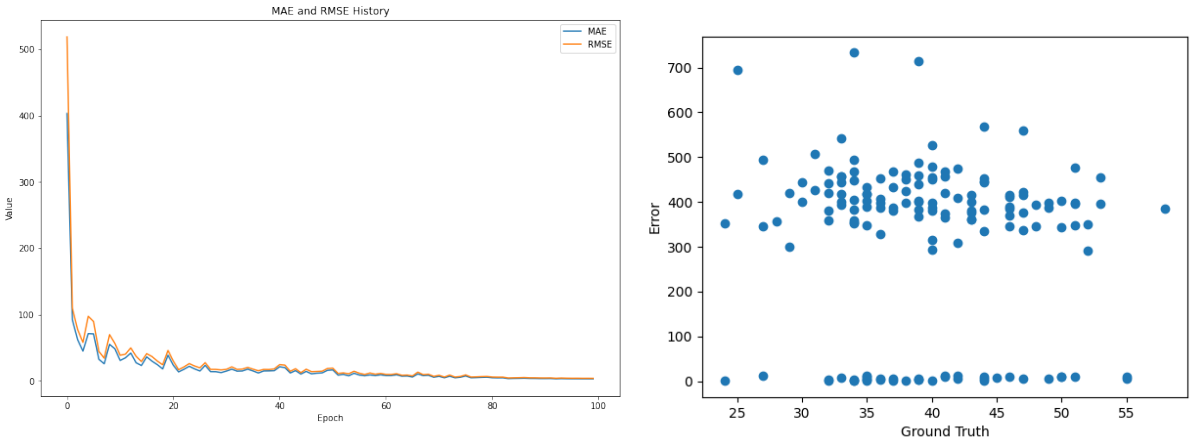

Figure 3: Paper[1] Implementation Train/Test Error

6 Additional studies

As previously mentioned, we run a deep analysis on the model's parameter to better fit the architecture to our data. In the specific our study aimed at finding both better data augmentation steps as well as losses and learning schedulers (tab. 6).

Loss	LR scheduler	Spread	MAE (0/3 shots)	
Masked L2	Cosine	No	3.12	3.00
Masked L2	Cosine	Yes	2.08	2.10
L1 + L2	Multiplicative	Yes	19.15	19.16
L1 + L2	ReduceOnPlateau	Yes	11.33	6.34
L1 + L2	Cosine	Yes	5.90	5.88
masked L1 + L2	Cosine	Yes	4.71	4.04

For what regards the loss we believe the original (implemented in CounTR code) loss to be the best, because it neglects noise and therefore allows the model to not be influenced by it. For the learning rate scheduler we believe our choice of Cosine Annealing to be better than the original scheduler if not the best as it stops the learning process from getting stuck and effectively is what allowed us to prevent the original model from overfitting our data (fig. 4, fig. 5).

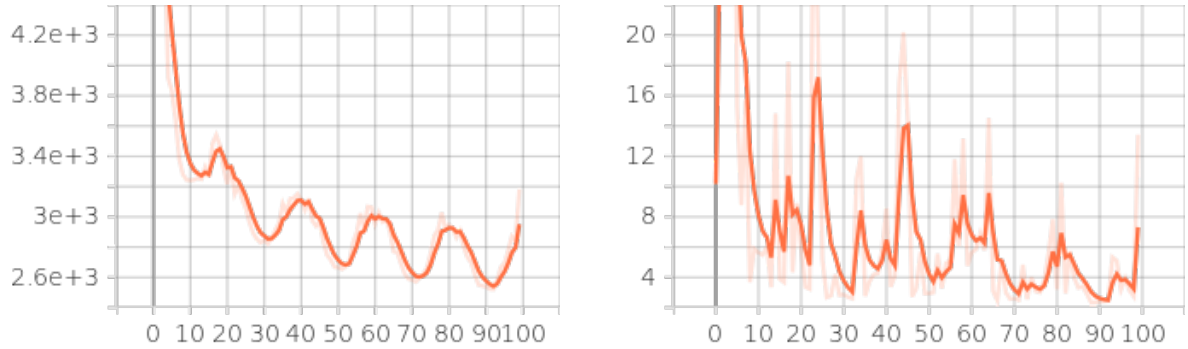

Figure 4: Loss and MAE curves with Cosine Annealing Scheduler

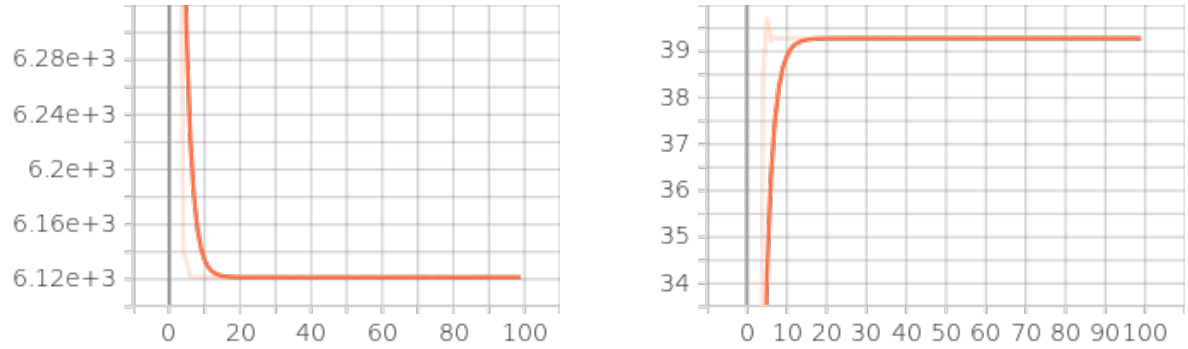

Figure 5: Loss and MAE curves with MultiplicativeLR Scheduler

7 Conclusions

Finally, the studies and experiments we conducted on these two different domains allow us to say that this architecture is capable of overcoming the limitations of a ViT. In literature it is known how ViTs have difficulties handling patches smaller than 16x16, but for most of the time in our two datasets we are dealing with even smaller entities. It is important to say that each of these applications require case-specific tuning of hyperparameters, such as the choice of a more complex learning scheduler. Putting this aside, we can confidently say that it is possible to finetune the model to solve tasks that were not taken into account when it was created.

For future improvements, we believe that this study might be extended to include exploring attention at each layer, since having knowledge of this information might help us understand which parameters truly matter when dealing with smaller patches. Concluding, we believe our Galaxy Counting Dataset to be usable (with some refinement) by observatories to keep track of the amount of galaxies at each point in time (and space).

References

[1] Chang Liu, Yujie Zhong, Andrew Zisserman, and Weidi Xie. Countr: Transformer-based generalised visual counting, 2022.

[2] Roberto E. González, Roberto P. Muñoz, and Cristian A. Hernández. Galaxy detection and identification using deep learning and data augmentation, 2018.