Partial-Attribution Instance Segmentation for Astronomical Source Detection and Deblending

Ryan Hausen
Department of Computer Science and Engineering
University of California, Santa Cruz
Santa Cruz, CA 95064
rhausen@ucsc.edu

Brant Robertson
Department of Astronomy and Astrophysics
University of California, Santa Cruz
Santa Cruz, CA 95064
brant@ucsc.edu

Abstract

Astronomical source deblending is the process of separating the contribution of individual stars or galaxies (sources) to an image comprised of multiple, possibly overlapping sources. Astronomical sources display a wide range of sizes and brightnesses and may show substantial overlap in images. Astronomical imaging data can further challenge off-the-shelf computer vision algorithms owing to its high dynamic range, low signal-to-noise ratio, and unconventional image format. These challenges make source deblending an open area of astronomical research, and in this work, we introduce a new approach called Partial-Attribution Instance Segmentation that enables source detection and deblending in a manner tractable for deep learning models. We provide a novel neural network implementation as a demonstration of the method.

1 Introduction

Astronomical images can contain tens of thousands of stars and galaxies (sources). Forthcoming telescopes including the Vera Rubin Observatory [11, 12], James Webb Space Telescope [26], and Nancy Grace Roman Space Telescope [24, 25] will push the current limits of observational astronomy and dramatically increase the number of sources to analyze. To measure accurate properties for these sources, we must *detect* sources by identifying statistically significant local maxima in an image and *deblend* sources by isolating the potentially overlapping flux distributions of each object. Consider a background-subtracted astronomical image $\mathcal{I} \in \mathbb{R}^{h \times w \times b}$ in which n sources are observed, where h is the height, w is the width, and b indicates the number of astronomical passbands. The image $\mathcal{I}$ can be decomposed into a sum of individual object contributions as

$$\mathcal{I} = \sum_{i=1}^{N} S_i + \epsilon \tag{1}$$

where $S_i \in \mathbb{R}^{h \times w \times b}$ represents the flux contributed to $\mathcal{I}$ by source i, and $\epsilon \in \mathcal{N}(0, \sigma)$ is the approximate noise distribution in the image. The process of decomposing an image into the form of Equation 1 represents the core challenge of source deblending. This submission presents a deep learning-based method to perform detection and deblending on astronomical images.

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

Table 1: Detection and deblending method categorization

Name	Detection Capacity	Deblend Type
SExtractor[2]	N	Disjoint
Morpheus[7]	N	Disjoint
Mask R-CNN[4]	N	Intersecting/Discrete
`blend2mask2flux`[3]	2	Intersecting/Discrete
Modified SRGAN[22]	0	Intersecting/Continuous
SCARLET[19]	0	Intersecting/Continuous
This Work	N	Intersecting/Continuous

1.1 Related Work

Source detection and deblending are well-studied problems in astronomy, and many approaches have been developed. Below, we highlight some popular and recent methods for source detection and deblending and point the interested reader to the review by Masias et al. [18].

Detection and deblending methods can be characterized by their *detection capacity* and *deblend type*. The detection capacity represents the number of sources a method can detect within a single image. For Equation 1, a detection capacity of N would indicate that a method could detect all sources appearing in an image. The deblend type indicates how the flux in a single pixel may be split between overlapping (blended) sources. A *disjoint* deblender assigns all flux in a pixel to a single source exclusively. An *intersecting/discrete* deblender can assign the flux to more than one source with uniform weighting across pixels. Finally, an *intersecting/continuous* deblender can assign the flux to more than one source with variable weighting across pixels.

Astronomical analysis methods vary in their detection and deblending methods. Bertin & Arnouts [2] introduced SExtractor that uses a convolution and thresholding approach for detection, and an isophotal analysis using binned pixel intensity for deblending. Hausen & Robertson [7] introduced Morpheus, a U-Net [23] style convolutional neural network (CNN) model that filters out background pixels, uses a thresholding approach for detection, and combines watershed and peak finding algorithms for deblending. Another U-Net based model called `blend2mask` [3] performs detection and deblending using the U-Net alone. Reiman & Göhre [22] use a modified Super-Resolution GAN (SRGAN) [16] to deblend overlapping sources. Burke et al. [4] trained a Mask R-CNN [8] model to detect and deblend sources. SCARLET [19] deblends sources using constrained matrix factorization.

Table 1 summarizes the features of these previous methods, none of which have a detection capacity of N and an *intersecting/continuous* deblend type. We now present a deep learning-based *intersecting/continuous* deblending algorithm with a detection capacity of N.

2 Partial-Attribution Instance Segmentation

Partial-Attribution Instance Segmentation (PAIS) is a new extension of the instance segmentation paradigm that allows for weighted, overlapping segmentation maps. PAIS differs from other segmentation schemes like cell segmentation [28], interacting surface segmentation [27], and amodal instance segmentation [17]. PAIS aims to isolate objects appearing in an image while preserving their measurable quantities within areas of overlap. For PAIS, we can approximate Equation 1 as

$$\tilde{\mathcal{I}} = \sum_{i=1}^{N} M_i \odot \mathcal{I} \tag{2}$$

where $\tilde{\mathcal{I}} \in \mathbb{R}^{h \times w \times b}$ estimates the background-subtracted flux image $\mathcal{I}$ in Equation 1, $M_i \in [0,1]^{h \times w \times b} st. \sum_i^N M_{i,jkl} = 1$ constitutes the pixel-level fractional contribution of source i to $\mathcal{I}$, and $\odot$ symbolizes the Hadamard product. Equation 2 is tractable for deep learning models, allowing the model to learn the bounded quantities M_i rather than the unbounded source images S_i. The N number of sources setting the upper limit of the sum in Equation 2 can differ for each image.

To construct a PAIS format that can be represented by a CNN, we have to construct an encoding for the M_i in Equation 2. Inspired by Cheng et al. [5] and Kendall et al. [13], we propose an encoding for

Table 2: *Partial Contribution Representation* encoding efficacy

Test	Value
Total Source Flux [e/s] (MAE)	1.97 ± 15.43
Two-Sample KS Test p-value	0.93 ± 0.22

the M_i components called *Partial Contribution Representation* (PCR). The goal of PCR is to encode, for any single pixel (j, k, l), the fractional contribution to its intensity from the closest n sources. Using PCR, a variable number N of sources can be encoded per image. PCR consists of three tensors: the Center-of-mass $C^c \in \{0, 1\}^{h \times w}$, Contribution-vectors $C^v \in \mathbb{R}^{h \times w \times n \times 2}$ and Contribution-maps $C^m \in [0, 1]^{h \times w \times b \times n}$. The center-of-mass encodes the locations of all the sources in an image. For any pixel, we set $C^c_{jk} = 1$ if that location indicates the center of a source and $C^c_{jk} = 0$ otherwise. The contribution-vector C^v_{jk} encodes the Cartesian offset to the closest n sources. The contribution-map C^m_{jkl} connects the fractional contribution of the n sources with the associated contribution-vectors C^v_{jk}. The fixed dimensionality of C^c, C^v, and C^m make PCR tractable for deep learning algorithms.

3 Our Approach

Our approach consists of making a PAIS dataset leveraging PCR and is implemented using a novel neural network architecture. We summarize our dataset, model, and training method below.

3.1 Dataset

To generate the PAIS input samples, we used the Hubble Legacy Fields (HLF) GOODS-South F160W (1.6μm) flux images [10], along with the 3D-HST source catalog [20]. The HLF images were split into training and test sets of 256×256 pixel subregions, with 2,000 training samples and 500 test samples. The input labels, as described in Section 2, consist of the center-of-mass images C^c, the contribution-vectors C^v, and the contribution-maps C^m. The center-of-mass training images are generated in a manner similar to Cheng et al. [5], by placing pixelated 2D Gaussians with standard deviation $\sigma = 8$(pixels) at the locations of sources in the 3D-HST catalog. The contribution-vectors, an extension to the method by Cheng et al. [5], are generated from the Cartesian offset to the nearest $n = 3$ sources to each pixel. The contribution-maps require the M_i values from Equation 2. To determine M_i, we use SCARLET [19] with the F125W, F160W, F606W, and F850LP flux and weight images from the HLF GOODS-South data and the TinyTim point-spread functions [15] to deblend the sources from the 3D-HST catalog. We then use PCR to encode the M_i from SCARLET. The complete dataset generation routine can be found in our project repository (`https://github.com/ryanhausen/morpheus-deblend/`).

To evaluate the efficacy of PCR to encode M_i, we define two metrics. We use the mean difference between the total flux determined by the SCARLET encoded M_i for each input source and that recovered by our encoding. We also use a two-sample Kolmogorov–Smirnov (KS) test to compare the normalized cumulative surface brightness profile within the radius encompassing 90% of the total flux of each source to evaluate the encoding of the spatial flux distribution. Table 2 reports the results and demonstrates that PCR encoding approximately preserves both the total flux and the spatial flux distribution for each source. With this verification, we can train a network to recover the PCR encoding for each input HST F160W image.

3.2 Model

To recover the PCR for training images, we developed a novel neural network architecture inspired by Cheng et al. [5], based on the Fast Attention Network [9] and implemented in TensorFlow [1]. The model features two decoders that share a single encoder. The first decoder, called the *spatial decoder*, predicts values for C^c and C^v. The second decoder, called the *attribution decoder*, predicts values for C^m. The complete model code can be found in the repository for this project (`https://github.com/ryanhausen/morpheus-deblend/`). An end-to-end example of the model can be seen in Figure 1.

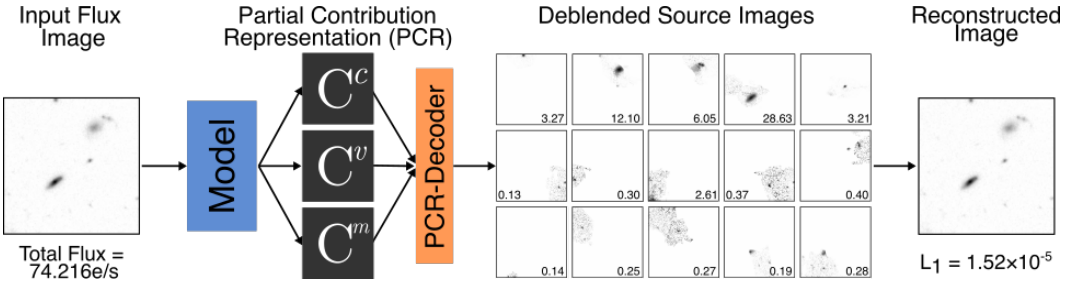

Figure 1: End-to-end example using our method to detect and deblend sources. Starting from the left: A flux image is input to the Model (see Section 3.2). The Model outputs the deblended image in the *Partial Contribution Representation* (PCR; see Section 2).The output from the model is then decoded using the non-learned PCR Decoder algorithm into separate deblended source images. The deblended source images have their total flux within r_{90} annotated. The deblended source images are then added together to generate the reconstructed image which has an L_1 total flux difference of 1.52×10^{-5} with the original input image.

Table 3: Training metric results

Metric	Training	Test
MAE	27.0183 ± 1.0658	28.5090 ± 0.3386
MSE	0.0114 ± 0.0001	0.0124 ± 0.0006
cross-entropy	0.9485 ± 0.0069	1.0806 ± 0.0098

3.3 Training

To train the model to recover the PCR of the input images, we use the Adam Optimizer [14] with a learning rate of 5×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.9999$, $\epsilon = 1 \times 10^{-7}$, and a batch size of 100. The model was trained for 1000 epochs using an NVIDIA V100 32GB GPU, taking 31 hours. The loss function for the model is composed of three functions. The *spatial decoder* outputs for C^c and C^v are penalized according to mean squared error (MSE) and the mean absolute error (MAE), respectively. The *attribution decoder* output C^m is penalized using cross-entropy loss with an additional entropy regularization term. In practice, we found that the additional entropy regularization helped incentivize the network to learn information about multiple sources in C^m. Each loss term is weighted and combined into a single loss function described by

$$\mathcal{L}_{\text{total}} = \lambda_{C^c}\mathcal{L}_{C^c} + \lambda_{C^v}\mathcal{L}_{C^v} + \lambda_{C^m}\mathcal{L}_{C^m} + \lambda_S\mathcal{L}_S, \tag{3}$$

where $\mathcal{L}_{C^c}$ is the MSE loss calculated between the model output and input label with $\lambda_{C^c} = 15$, $\mathcal{L}_{C^v}$ is the MAE loss calculated between the model output and input label with $\lambda_{C^v} = 0.06$, $\mathcal{L}_{C^m}$ is a cross-entropy loss calculated between the model output and input label with $\lambda_{C^m} = 4$, and $\mathcal{L}_S$ is the entropy regularization on the model C^m output with $\lambda_S = 2$. See Table 3 for a summary of the training results, demonstrating a good balance between test and training error. A complete log of training experiments is available at (`https://www.comet.ml/ryanhausen/morpheus-deblend/`).

4 Discussion and Future Work

In this work, we introduced the Partial Attribution Instance Segmentation (PAIS) scheme for astronomical source debelending. We presented *Partial Contribution Representation* (PCR) as a method for implementing PAIS within deep learning-based models. We demonstrated the efficacy of PCR for encoding the results of existing astronomical deblenders, and developed a novel neural network architecture to recover the PCR from input flux images. While we demonstrated deblending for single band (F160W) images, PCR can be extended to multiband images. As with many supervised

methods, our model requires labeled training data. To apply this method on other survey datasets may require the use of transfer learning [6, 21] or retraining.

Acknowledgments and Disclosure of Funding

5 Acknowledgements

RDH would like to thank Roberto Manduchi for helpful conversations. BER acknowledges support from NASA contract NNG16PJ25C and grant 80NSSC18K0563. The authors acknowledge use of the lux supercomputer at UC Santa Cruz, funded by NSF MRI grant AST 1828315.

6 Broader Impact

This work develops a novel method for separating source signals in astronomical images. Due to the specialized format and problem setting, the authors do not see any broader negative societal impacts as a result of this work.

References

[1] Abadi, M., Agarwal, A., Barham, P., et al. 2016, arXiv e-prints, arXiv:1603.04467. https://arxiv.org/abs/1603.04467

[2] Bertin, E., & Arnouts, S. 1996, A&AS, 117, 393, doi: 10.1051/aas:1996164

[3] Boucaud, A., Huertas-Company, M., Heneka, C., et al. 2020, MNRAS, 491, 2481, doi: 10.1093/mnras/stz3056

[4] Burke, C. J., Aleo, P. D., Chen, Y.-C., et al. 2019, MNRAS, 490, 3952, doi: 10.1093/mnras/stz2845

[5] Cheng, B., Collins, M. D., Zhu, Y., et al. 2019, arXiv e-prints, arXiv:1911.10194. https://arxiv.org/abs/1911.10194

[6] Domínguez Sánchez, H., Huertas-Company, M., Bernardi, M., et al. 2019, MNRAS, 484, 93, doi: 10.1093/mnras/sty3497

[7] Hausen, R., & Robertson, B. E. 2020, ApJS, 248, 20, doi: 10.3847/1538-4365/ab8868

[8] He, K., Gkioxari, G., Dollár, P., & Girshick, R. 2017, arXiv e-prints, arXiv:1703.06870. https://arxiv.org/abs/1703.06870

[9] Hu, P., Perazzi, F., Caba Heilbron, F., et al. 2020, arXiv e-prints, arXiv:2007.03815. https://arxiv.org/abs/2007.03815

[10] Illingworth, G., Magee, D., Bouwens, R., et al. 2016, arXiv e-prints, arXiv:1606.00841. https://arxiv.org/abs/1606.00841

[11] Ivezić, Ž., Kahn, S. M., Tyson, J. A., et al. 2008, arXiv e-prints. https://arxiv.org/abs/0805.2366

[12] Ivezić, Ž., Kahn, S. M., Tyson, J. A., et al. 2019, ApJ, 873, 111, doi: 10.3847/1538-4357/ab042c

[13] Kendall, A., Gal, Y., & Cipolla, R. 2017, arXiv e-prints, arXiv:1705.07115. https://arxiv.org/abs/1705.07115

[14] Kingma, D. P., & Ba, J. 2014, ArXiv e-prints. https://arxiv.org/abs/1412.6980

[15] Krist, J. E., Hook, R. N., & Stoehr, F. 2011, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 8127, Optical Modeling and Performance Predictions V, ed. M. A. Kahan, 81270J, doi: 10.1117/12.892762

[16] Ledig, C., Theis, L., Huszar, F., et al. 2016, arXiv e-prints, arXiv:1609.04802. https://arxiv.org/abs/1609.04802

[17] Li, K., & Malik, J. 2016, arXiv e-prints, arXiv:1604.08202. https://arxiv.org/abs/1604.08202

[18] Masias, M., Freixenet, J., Lladó, X., & Peracaula, M. 2012, MNRAS, 422, 1674, doi: 10.1111/j.1365-2966.2012.20742.x

[19] Melchior, P., Moolekamp, F., Jerdee, M., et al. 2018, Astronomy and Computing, 24, 129, doi: 10.1016/j.ascom.2018.07.001

[20] Momcheva, I. G., Brammer, G. B., van Dokkum, P. G., et al. 2016, ApJS, 225, 27, doi: 10.3847/0067-0049/225/2/27

[21] Pratt, L. Y. 1993, in Advances in Neural Information Processing Systems 5, ed. S. J. Hanson, J. D. Cowan, & C. L. Giles (Morgan-Kaufmann), 204–211. http://papers.nips.cc/paper/641-discriminability-based-transfer-between-neural-networks.pdf

[22] Reiman, D. M., & Göhre, B. E. 2019, MNRAS, 485, 2617, doi: 10.1093/mnras/stz575

[23] Ronneberger, O., Fischer, P., & Brox, T. 2015, arXiv e-prints, arXiv:1505.04597. https://arxiv.org/abs/1505.04597

[24] Spergel, D., Gehrels, N., Breckinridge, J., et al. 2013, arXiv e-prints, arXiv:1305.5422. https://arxiv.org/abs/1305.5422

[25] Spergel, D., Gehrels, N., Baltay, C., et al. 2015, arXiv e-prints, arXiv:1503.03757. https://arxiv.org/abs/1503.03757

[26] Williams, C. C., Curtis-Lake, E., Hainline, K. N., et al. 2018, ApJS, 236, 33, doi: 10.3847/1538-4365/aabcbb

[27] Xie, H., Pan, Z., Zhou, L., et al. 2020, arXiv e-prints, arXiv:2007.01259. https://arxiv.org/abs/2007.01259

[28] Zhou, Y., Chen, H., Lin, H., & Heng, P.-A. 2020, arXiv e-prints, arXiv:2007.10787. https://arxiv.org/abs/2007.10787