Highlights of Spanish Astrophysics XI, Proceedings of the XV Scientific Meeting of the Spanish Astronomical Society held on September 4–9, 2022, in La Laguna, Spain. M. Manteiga, L. Bellot, P. Benavidez, A. de Lorenzo-Cáceres, M. A. Fuente, M. J. Martínez, M. Vázquez- Acosta, C. Dafonte (eds.), 2023

Astrometric Centering of WFPC2/HST images with Deep Learning.

Baena-Gallé, R.[1], Girard, T.M.[2], Casetti-Dinescu, D.I.[2], and Martone, M.[2]

[1] Universidad Internacional de la Rioja. Avenida de la Paz, 137, 26006 Logroño, La Rioja, Spain. https://orcid.org/0000-0001-5214-7408
[2] Physics Dept., Southern Connecticut State University, 501 Crescent Street, New Haven CT 06515, USA

Abstract

Archival WFPC2/HST exposures hold great potential for proper motion studies. Up to now, the astrometric precision of WFPC2 images is limited to $\sim$ 20 mpix as these are the most undersampled images from the various HST optical imagers. We explore deep-learning techniques, specifically, Convolutional Neural Network (CNN) algorithms, to implicitly model the PSF and determine stars centers. The method is tested on HST images and the resulting astrometric precision is compared to that obtained with traditional, state-of-the-art centering algorithms. This approach is data-driven as the behaviour of CNNs is not based on a predefined PSF, rather it is based on estimating the stars' center positions directly from pixel intensity values in the images. We present the description of the CNN architecture, data preprocessing, and learning strategy together with preliminary results from simulated WFPC2 data.

1 Introduction

The study of stars' proper-motions allows us to go deeper into our understanding of how the local universe is evolving. In this respect, the long temporal baseline provided by the Wide Field Planetary Camera 2 (WFPC2), at Hubble Space Telescope (HST), is of high importance ([4], [5], [8]). For example, the Mikulski Archive for Space Telescope includes a rich WFPC2 database of around one hundred globular clusters in the Milky Way and other regions near the Magellanic Clouds. However, performing precision astrometry is essential for this task, taking into account that the average proper motion of regular stars is of the order of milli-arcsec and the pixel resolution of the WF chip is 0.1 arcsec/pixel (0.045 arcsec/pixel for PC).

Traditionally, the problem of estimating stars' proper-motions has been faced by fitting a predefined PSF shape to every star, and using some centering algorithm to estimate its intra-pixel position [6]. Here, we share first results of a new approach based on Deep Learning (DL) methodology that does not make any assumption of the PSF morphology. Specifically,

we train a Convolution Neural Network (CNN) model, in a supervised manner over a set of point-like simulated stars, to measure correlations between pixels. Thus, the way the star light is distributed across the field of view (FOV) provides a measurement of the star's position within the pixel.

2 Classical Centering Algorithms

All classic methods used to date basically differ in the PSF and the centering algorithm to provide star positions at milli-pixel precision. Among those tested in [6] are as follows:

1. 2D elliptical Gaussian and centering routines developed by [11]. This method is fast and works relatively well on bright stars.
2. Effective PSFs (ePSF) built from real observations, which are the ones used in the hst1pass code developed by [1]. This method is especially good at low SNR but it has the tendency to discard sources close to the saturation level.
3. PSFs created with ray tracer Tiny Tim [10] and Dolphot v2.0 package developed by [7]. This algorithm is optimized for point-like sources but is computationally expensive.
4. PSFs computed empirically from real images to take into account the variability accross the field of view (FOV) by means of SExtractor software (PSFex). It uses a set of 636 exposures (160s. at filter F555W) from globular cluster 47 Tucanae. SExtractor is also used to compute centering measurements [3]
5. The image is deconvolved in the Fourier domain prior to the computation of star's centers. An ideal PSF at the instrument diffraction limit is used, as well as a LP Butterworth filter to avoid noise amplification. The resulting image is processed with the method from [11]. Sources with low SNR are usually lost.

In general, it can be stated that the method using a library of ePSFs within the code hst1pass [1] yields the best results, at the cost of discarding sources near the saturation regime. We refer the reader to [6] for further details.

3 Deep Learning Model for Centering

To our knowledge, Deep Learning (DL) has never been used for astrometric estimations in stars' proper-motion studies with HST data. Our approach does not make any assumption of the PSF shape, but it estimates the (x,y) coordinates of the star center by measuring correlations in the pixel intensity values within an aperture around the star.

Our particular CNN model is a VGG [12] with six trainable layers, four of them convolutional layers plus two fully-connected which end in two neurons outputs, each of them to provide an estimate in x- and y-axis, respectively. One max-pool layer is inserted every two

convolutional ones, and all hidden layers are equipped with ReLu non-linearity except the last one which is *linear*, in keeping with the regression nature of this problem. We found that inserting a batch-normalization layer after the fourth convolutional one yields better results and helps stabilize the convergence process in our particular problem. The final architecture is plotted in Fig. 1, subplot a.

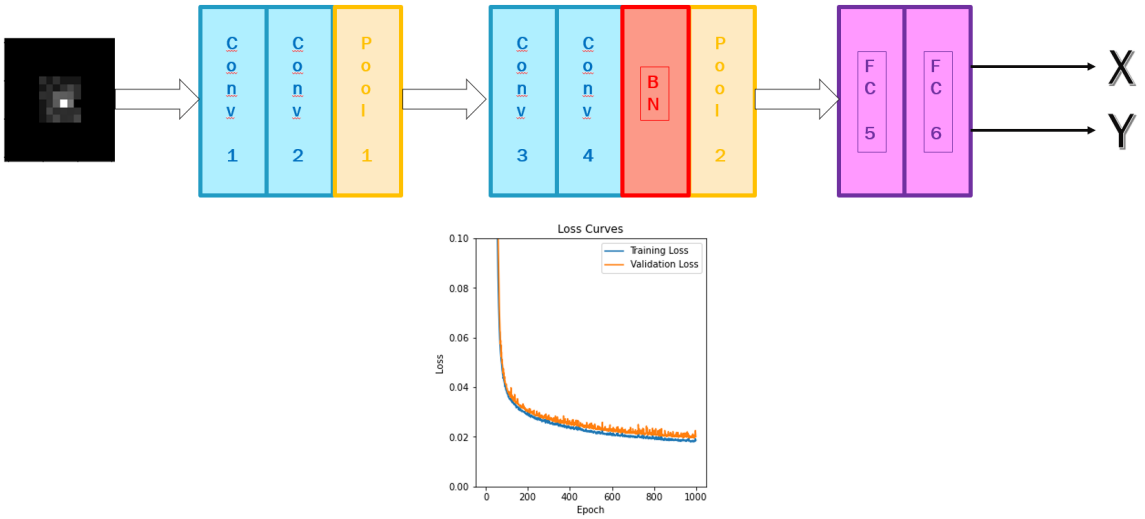

Figure 1: Top: final VGG model. Bottom: loss curves for training and validation dataset.

Due to the finite dimensions of each cutout image of 6×6 pixels around a star, the model is limited in depth. Thus, we cannot add up as many layers as we want, a feature that is generally assumed to increase the level of detail that can be analysed or the number of features that can be extracted from the data. Hence, we opted to modify the number of trainable parameters by increasing the number of kernels at each layer. This allows us to check how the model behaves with respect to the number of parameters in terms of overfitting. This is critical at this stage since our simulations are based on shift-invariant PSFs and isolated sources with no light contamination from nearby sources. Hence, two different VGG models are trained, the first one with $34K$ parameters, and the second one with $214K$. We found that the second one exhibits better results but with some overfitting effect during training. Images are also framed with a zero padding to guarantee a minimum number of layers as the model increases in depth. A typical training process is plotted in Fig. 1, subplot b.

The dataset consists of $\sim$ 4,600 images of individual stars which are divided in $70-10-20\%$ for training, validation and final test. All images are normalized to sum one, independently of the noise level or if the star is saturated. All stars are located over the same image pixel, hence, output positions are also normalized between 0 and 1 so the model is estimating relative shifts within the same pixel. Batch sizes are between 250 and 325 images. We noticed the minimization of the cost function (Mean Absolute Error) benefits of low learning rates (10^{-5}), while epochs had to be increased (between 1,000 and 2,000). The model was designed in Keras/TF and optimized with ADAM using default values.

4 Dataset description and Results

The Deep Learning (DL) model is tested over images of point-like sources which simulate typical observations with the PC and WF chips of the WFPC2 camera. Two different simulation codes were used to generate the data set. The first one, SkyMaker [2], makes use of a user defined PSF model, a list of sky positions, and CCD specifications, which vary depending on the pixel size and the noise level. From a whole image simulation, for instance, a cluster, a set of individual stars can be extracted with known positions. The second one, currently under development and from now on named Simu_WFPC2, is being created specifically for the PC/WF3 chips. The code includes, among other specifications, the variability of the PSF across the chip, vignetting, charge transfer inefficiency effects, CCD columns illuminations due to electron excess, A/D saturation, background, cosmic rays, bias, darks, shot noise, etc... Hence, its simulated images can be considered as more realistic than the ones provided by SkyMaker. At the current stage of this project, we are not making use of the variability of the PSF over the chip. Both simulations were fed with two different PSF models, the ePSFs used by the library in the hst1pass code and the PSFex computed by SExtractor from a real dataset, namely exposures in the field of NGC 104. The former can be considered somehow more realistic than the latter. In all cases, we have a "ground truth" of star positions and magnitudes which allows us to train the model in a supervised way.

Table 1: MEAN RESIDUALS IN (x,y) FOR PC (top) AND WF (bottom)

Simulation + PSF	[11]	[1]	VGG 34K	VGG 214K
Simu_WFPC2 + ePSF	(31.9 , 27.9)	(9.5 , 8.6)	(7.8 , 6.9)	(6.8 , 6.8)
Simu_WFPC2 + PSFex	(15.0 , 12.0)	(47.7 , 43.7)	(6.4 , 6.4)	(5.4 , 5.4)
SkyMaker + ePSF	(36.3 , 32.1)	(8.7 , 8.5)	(9.3 , 8.1)	(7.5 , 6.4)
SkyMaker + PSFex	(17.3 , 13.6)	(48.5 , 41.7)	(8.5 , 7.5)	(6.8 , 6.3)
Simulation + PSF	[11]	[1]	VGG 34K	VGG 214K
Simu_WFPC2 + ePSF	(34.1 , 35.1)	(8.3 , 8.0)	(7.8 , 7.4)	(6.8 , 6.5)
Simu_WFPC2 + PSFex	(12.3 , 13.7)	(13.6 , 14.3)	(6.7 , 6.0)	(5.0 , 5.0)
SkyMaker + ePSF	(41.1 , 44.3)	(9.1 , 9.0)	(8.7 , 8.8)	(7.4 , 7.2)
SkyMaker + PSFex	(13.4 , 15.8)	(20.8 , 16.2)	(8.3 , 7.4)	(5.9 , 5.9)

Table 1 shows mean residuals in both x- and y-axis for PC and WF3 chip achieved by two classic methods [11] and [1], as well as for the DL model with two different numbers of trainable parameters. Outliers above 3σ have been removed. All four methods were applied on the same set of stars and known positions, simulated by SkyMaker and Simu_WFPC2 and making use of ePSF and PSFex. It is shown that the VGG with 214K parameters exhibit slightly better results w.r.t. hst1pass, in particular when ePSF was used to generate the data set. Additionally, DL models are more stable in their results independently of the simulation code and the PSF used to generate the dataset.

Figure 2 plots for all four methods the distribution of x, y-residuals (left panels) and their

dependence with magnitude (middle and right panels). Residuals shown are for the PC chip under the configuration Simu_WFPC2 + ePSF. Similar results are obtained for the WF chip. The hst1pass code has a built-in quality that can be used to reject outlier star images thus producing tighter plots. On the other hand, hst1pass cannot provide results for magnitudes < 15, in contrast with the DL model, which extends this range up to magnitudes < 14.

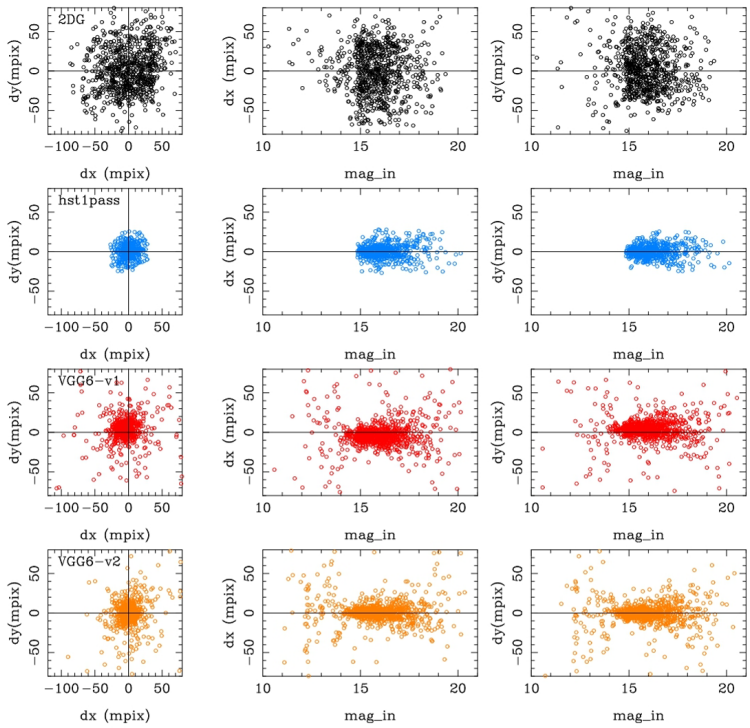

Figure 2: From top to bottom: 2D elliptical Gaussian and centering routine by [11]; ePSFs and hst1pass by [1]; VGG with 34K parameters; VGG with 214K parameters. Leftmost column: residuals distribution w.r.t. (x,y) position. Middle and right columns: residuals in x,y as a function of magnitude. Simulations are with Simu_WFPC2 and using ePSFs. Results are for the PC chip. Similar results are obtained for WF.

5 Conclusions and Future Work

The VGG model introduced in this paper has performed well on simulated HST data, overcoming some challenges of traditional centering methods and performing comparably or slightly better that the state-of-the-art classic algorithm in hst1pass. Our current simulations can still be improved to introduce non-isolated stars and the variation of the PSF across the FOV. This can limit the final performance of the current model, taking into account that we are constrained in the number of convolutional layer, due to the input image size, and the number of trainable parameters, due to the overfitting effect. We have recently

begun testing this methodology on real WFPC2 data. Preliminary results are encouraging.

Finally, we are planning to test a new model on this problem, in particular, ResNets architectures [9]. These architectures involve skipping connections which connects activations of a layer to further layers by skipping some of them in between, thus forming a so-called residual block. ResNets are built by stacking these residual blocks together. Therefore, the final network will fit the residual mapping instead of the features mapping, resulting in training deeper networks avoiding vanishing gradients. Additionally, we will explore the use of larger image sizes by upsampling as well as an automatic choice of the training hyperparameter.

Acknowledgments

RBG was funded by the Universidad Internacional de la Rioja (UNIR) Research Project "ADELA: Aplicaciones de Deep Learning para Astrofísica", no. B0036, and by the Call for Grants for Research Stays Abroad 2021/2022 of UNIR. TG, DC and MM are funded by the NASA Connecticut Space grant 80NSSC20M0129.

References

[1] Anderson, J., & King, I. R. 2000, PASP, 112, 1360

[2] Bertin, E. 2009, Memorie della Società Astronomica Italiana, 80, 422

[3] Bertin, E., & Arnouts, S. 2010, SExtractor: Source Extractor, AstrophysicsSource Code Library, record ascl:1010.064

[4] Casetti-Dinescu, D., Girard, T. M., Jílková, L., van Altena, W. F., Podestá, F. & López, C. 2013, AJ, 146, 33

[5] Casetti-Dinescu, D., Girard, T. M. & Schriefer, M. 2017, MNRAS, 473, 4064

[6] Casetti-Dinescu, D., Girard, T. M., Kozhurina-Platais, V., Platais, I., Anderson, J. & Horch, E. P. 2021, PASP, 133, 064505

[7] Dolphin, A. E. 2000, PASP, 112, 1383

[8] Girard, T. M., van Altena, W. F., Zacharias, N., Vieira, K., Casetti-Dinescu, D., Castillo, D. J., Herrera, D., Lee, Y. S., Beers, T. C., Monet, D. G. & López, C. E. 2011, AJ, 142, 15

[9] K. He, X. Zhang, S. Ren & J. Sun. 2016 Proc. CVPR, pp. 770-778

[10] Krist, J. E., Hook, R. N., & Stoehr, F. 2011, Proc. SPIE, 8127, 81270J

[11] Lee, J.-F., & van Altena, W. F. 1983, AJ, 88, 1683

[12] Simonyan, K. & Zisserman, A. 2015, Proc. ICLR, arXiv preprint arXiv:1409.1556