



**HAL**  
open science

## Deep learning for galaxy surface brightness profile fitting

D. Tuccillo, M. Huertas-Company, E. Decenci re, S. Velasco-Forero, H. Dom nguez S nchez, P. Dimauro

► **To cite this version:**

D. Tuccillo, M. Huertas-Company, E. Decenci re, S. Velasco-Forero, H. Dom nguez S nchez, et al.. Deep learning for galaxy surface brightness profile fitting. Monthly Notices of the Royal Astronomical Society, 2018, 475, pp.894-909. <10.1093/mnras/stx3186>. <insu-03718938>

**HAL Id: insu-03718938**

**<https://insu.hal.science/insu-03718938v1>**

Submitted on 10 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destin e au d p t et   la diffusion de documents scientifiques de niveau recherche, publi s ou non,  manant des  tablissements d'enseignement et de recherche fran ais ou  trangers, des laboratoires publics ou priv s.



HAL Authorization

# Deep learning for galaxy surface brightness profile fitting

D. Tuccillo,<sup>1,2★</sup> M. Huertas-Company,<sup>1,3,4</sup> E. Decencière,<sup>2</sup> S. Velasco-Forero,<sup>2</sup>  
H. Domínguez Sánchez<sup>1,3</sup> and P. Dimauro<sup>1</sup>

<sup>1</sup>LERMA, Observatoire de Paris, PSL Research University, CNRS, Sorbonne Universités, UPMC Univ. Paris 06, F-75014 Paris, France

<sup>2</sup>MINES Paristech, PSL Research University, Centre for Mathematical Morphology, Fontainebleau, France

<sup>3</sup>Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA 19104, USA

<sup>4</sup>University of Paris Denis Diderot, University of Paris Sorbonne Cité (PSC), F-75205 Paris Cedex 13, France

Accepted 2017 December 4. Received 2017 December 4; in original form 2017 September 24

## ABSTRACT

Numerous ongoing and future large area surveys (e.g. Dark Energy Survey, EUCLID, Large Synoptic Survey Telescope, Wide Field Infrared Survey Telescope) will increase by several orders of magnitude the volume of data that can be exploited for galaxy morphology studies. The full potential of these surveys can be unlocked only with the development of automated, fast, and reliable analysis methods. In this paper, we present *DeepLeGATo*, a new method for 2-D photometric galaxy profile modelling, based on convolutional neural networks. Our code is trained and validated on analytic profiles (*HST*/CANDELS F160W filter) and it is able to retrieve the full set of parameters of one-component Sérsic models: total magnitude, effective radius, Sérsic index, and axis ratio. We show detailed comparisons between our code and *GALFIT*. On simulated data, our method is more accurate than *GALFIT* and  $\sim 3000$  time faster on GPU ( $\sim 50$  times when running on the same CPU). On real data, *DeepLeGATo* trained on simulations behaves similarly to *GALFIT* on isolated galaxies. With a fast *domain adaptation* step made with the 0.1–0.8 per cent the size of the training set, our code is easily capable to reproduce the results obtained with *GALFIT* even on crowded regions. *DeepLeGATo* does not require any human intervention beyond the training step, rendering it much automated than traditional *profiling* methods. The development of this method for more complex models (two-component galaxies, variable point spread function, dense sky regions) could constitute a fundamental tool in the era of big data in astronomy.

**Key words:** methods: data analysis – catalogues – galaxies: general – galaxies: high-redshift – galaxies: structure.

## 1 INTRODUCTION

The characterization of the structure of galaxies inferred from their surface brightness distribution is a powerful tool in astronomy. The earliest studies on galaxy structural characterization lead to the discovery of the de Vaucouleurs profile (de Vaucouleurs 1958) for simple 1-D intensity profile fitting. Subsequently, the fitting law was generalized by Sérsic (1968) and increasingly complicated 1-D component fitting came in work by Kormendy (1977) and Kent (1985), where galaxies were decomposed into distinct components rather than into a single light profile.

Nowadays the description of the galaxy structure is often obtained with software of *profile fitting* (or *profiling*) that fit the surface light distribution of the galaxy with analytic functions (either parametric or non-parametric) in order to obtain a set of simple parameters that would ideally allow the reconstruction of the 2D photometric shape

of the galaxy. Computing structural parameters for large samples of galaxies allows us to derive more robust scaling relations at low and high redshift (Bernardi et al. 2013; van der Wel et al. 2014) as well as to test theoretical models. Studies of the scaling relations of different galaxy components also rely on robust and reproducible methods to measure and describe galaxy structure. The images provided by large-area surveys like the Sloan Digital Sky Survey (SDSS; York et al. 2000), the Galaxy And Mass Assembly (Driver et al. 2009), or at high redshift by CANDELS (Grogin et al. 2011; Koekemoer et al. 2011) have been effectively used to study the size distribution of galaxies and its dependence on their luminosity (Shen et al. 2003; Bernardi et al. 2013; Lange et al. 2016). The large data bases provide accurate statistics when investigating the distribution of mass and luminosity-surface brightness relation for different classes of galaxies (Driver et al. 2007; Kelvin et al. 2014; van der Wel et al. 2014; Kennedy et al. 2016).

The era of big data in astronomy is marked by the numerous current and future large area surveys like EUCLID, the Large Synoptic Survey Telescope (LSST Science Collaboration et al. 2009,

\* E-mail: [diego.tuccillo@obspm.fr](mailto:diego.tuccillo@obspm.fr)

the Wide Field Infrared Survey Telescope (WFIRST), Kilo-Degree Survey (de Jong et al. 2013), and Dark Energy Survey (DES). These surveys will increase by several orders of magnitude, in a few years, the volume of data that can be exploited for galaxy morphology studies, offering a unique opportunity to constrain models and infer properties of galaxies. In fact, the sheer number of galaxies available with morphological information and photometric or spectroscopic redshifts will allow precise studies of the rarest populations of active galactic nuclei (AGNs) and galaxies, like massive early-type galaxies at high redshift. Improving studies on the co-evolution of their multivariate distribution functions (luminosity, mass function, stellar mass, etc.). The large volume probed by the new surveys will also make it possible to map the small- and large-scale galactic environment at all redshifts, and to perform, at early cosmological epochs, a statistically significant analysis of the environment effect on the galaxy and AGN properties. The full potential of these surveys can be unlocked only with the development of automated, fast, and reliable methods to describe galaxy structure. The most popular galaxy fitting codes currently used in literature, i.e. GALFIT (Peng et al. 2002) and GIM2D (Simard et al. 2002), have not been conceived to deal with large amounts of data and several efforts have been made to automatize their use for catalogue compilation in large survey applications. GALAPAGOS, programmed by Barden et al. (2012), combines SExtractor (Bertin & Arnouts 1996) for source detection and extraction, and then makes use of GALFIT for modelling Sérsic profiles. GALAPAGOS has been proved to be robust in terms of parameter recoverability, however the results of the quality of the fitting depend heavily on the choice of the input parameters. With a similar concept, PYMORPH (Vikram et al. 2010) glues together GALFIT and SExtractor in a single pipeline written in PYTHON. The urgency created by the new generation of surveys has led to recent efforts to develop new fitting codes like ProFit (Robotham et al. 2017) programmed in C++ and directly conceived to be faster than older profile fitting codes, therefore exploitable for structural analysis of large amounts of data.

Deep learning has revolutionized data analysis in the last few years (LeCun, Bengio & Hinton 2015; Schmidhuber 2015). In the field of computer vision, convolutional neural networks (CNNs) have become the dominant approach for image processing and analysis (Krizhevsky, Sutskever & Hinton 2012). One of the main benefits of CNNs is that they learn representations automatically from raw inputs, recovering higher level features from lower-level ones, e.g. in images, the hierarchy of objects, parts, motifs, and local combinations of edges. In other words, whereas classical pattern recognition techniques need manual feature engineering to generate the outputs, deep learning automatically builds relevant descriptors from the pixels of the training set, not making any prior assumption on specific features of physical models of the specific problem. Another advantage is that deep learning, using distributed representations, combats the exponential challenges of the curse of dimensionality, making it extremely well suited to big data problems.

In astronomy, several groups have recently explored the application of deep learning methods. Most notably for morphological classification of galaxies, Dieleman, Willett & Dambre (2015) developed a CNN that reached accuracy >99 per cent in the classification of SDSS galaxies previously classified in the context of the Galaxy Zoo project (Lintott et al. 2008). A similar level of accuracy was obtained by Huertas-Company et al. (2015), who, using a CNN, retrieved for over 50 000 unclassified CANDELS FIELDS galaxies, the probabilities of having a spheroid or a disc presenting an irregularity, being compact or a point source. CNNs have also been used for star-galaxy classification (Kim & Brunner 2017) of SDSS data,

automated spectral feature extraction (Wang, Guo & Luo 2017), and unsupervised feature-learning for galaxy SEDs (Frontera-Pons et al. 2017).

In this work, we explore for the first time the possibility of applying CNNs for 2-D light profile galaxy-fitting. A deep learning approach to this problem may be extremely valuable for applications on large surveys, because it does not require any hand-made tuning previous to the application of the algorithm, thus automating the processes and greatly cutting the analysis times. In particular, we developed a CNN to decompose the galaxy structure of one-component *H*-band *HST*/CANDELS galaxies in terms of their total magnitude, radius, Sérsic index, and axis ratio. Although we obtain good and reliable results that may be already used for astronomical applications, we consider this work as a *proof of concept* on the concrete possibility to apply deep learning methods for this class of astronomical analysis.

The layout of the paper is as follows. In Section 2 we present the artificial image simulations and the real data that we use through this work, and in Sections 3 and 4 we describe our method and present our algorithm that we name *DeepLeGATo*. In Section 5, we compare the results obtained from *DeepLeGATo* and GALFIT on a sample of 5000 simulated galaxies. In Section 6 we discuss the *domain adaptation* of our code and we repeat the comparison between *DeepLeGATo* and GALFIT on a sample of 1000 real galaxies from the *HST*/CANDELS field. Finally, in Section 7, we discuss the conclusion of our work and the planned future development of the method here presented.

## 2 DATA

In this work we used two sets of data. A set of 55 000 images of artificially simulated *HST*/CANDELS -like galaxies (Section 2.2) and 5000 stamps of galaxies from *HST*/CANDELS (Section 2.1). In this section, we describe the simulations and the data, while in the following sections we will discuss how we used these data to train and test our profile fitting code.

### 2.1 Real data

The Cosmic Assembly Near-IR Deep Extragalactic Legacy Survey (CANDELS) is the largest project ever undertaken by the *HubbleSpaceTelescope* imaging data products and consist of five multiwavelength sky regions fields, each with extensive multi-wavelength observations. The core of CANDELS data consists of imaging obtained in the near-infrared by the Wide Field Camera 3 (WFC3/IR) camera, along with images in the visible-light obtained with the Advanced Camera for Surveys (ACS) camera. The CANDELS/Deep survey covers  $\sim 125$  square arcminutes within GOODS-N and GOODS-S, while the remaining consists of the CANDELS/Wide survey, achieving a total of  $\sim 800$  square arcminutes across GOODS and three additional fields (EGS, COSMOS, and UDS). A full description of the CANDELS observing program is given by Grogin et al. (2011) and Koekemoer et al. (2011).

We randomly selected 5000 *HST*/CANDELS galaxies, with the only condition of having magnitude, radius, Sérsic index, and axis ratio spanning in the range of the parameters used to build the simulated data (see Table 1).

For all selected galaxies, there are structural parameters available measured with GALFIT/GALAPAGOS from van der Wel et al. (2012), which is focused on the WFC3 data only, i.e. on the three filters F105W, F125W, and F160W. In particular, we select all the galaxies of our sample in the F160W filter. In this work we also make

**Table 1.** We simulated 55 000 realistic one-component F160W filter (*H* band) *HST*/CANDELS galaxies, having structural parameter within these ranges of values.

	Magnitude (AB)	Radius (pixels)	Sérsic index	Axis ratio
Range	16–23	1.5–31.6	0.3–6.2	0.2–0.8

use of the catalogue from (Dimauro et al. 2017, submitted), presenting the bulge-disc decompositions for the surface brightness profiles of 17.600 *H*-band selected galaxies in the CANDELS fields ( $F160W < 23$ ,  $0 < z < 2$ ) in 4–7 filters covering a spectral range of 430–1600 nm.

In Fig. 1 we show the distributions of the GALFIT-derived parameters of our sample. As we will discuss in the next sections, we used different subsets of real galaxies in order to test our code and to transfer the learning acquired on simulated data, to the case of real galaxies (process known as *domain adaptation*).

## 2.2 Simulated data

We simulated 55 000 galaxies using one-component Sérsic models. The algorithm that we used for the simulations is the 1.4 version of *GalSim* (Rowe et al. 2015), an open-source software whose bulk of the calculations is carried out in C<sup>++</sup>. In order to simulate realistic galaxy images, we used real point spread function (PSF) and real noise from the *HST*/CANDELS F160W filter (*H* band). Our data set was obtained uniformly varying magnitude, radius, axis ratio, Sérsic index, and position angle of the galaxies. All the stamps were generated in order to have a size of  $128 \times 128$  pixels and a pixel scale of 0.06 arcsec.

The total surface brightness of the galaxies is given from the integral over the galaxy area of the flux intensity distribution  $I(r)$  measured in units of luminosity per unit area at position  $(x, y)$ , i.e.  $2\pi \int I(r) r dr$ . As conventionally, we express the surface brightness in units of magnitudes per square arcsec, that is related to the physical surface brightness profile through  $\text{mag} \propto -2.5 \log_{10} I(r)$ . The radial surface brightness profile of a galaxy is described by the Sérsic function given by:

$$I(r) = I(0) \exp[-b_n (r/r_e)^{1/n}] \quad (1)$$

where  $r_e$  is the effective radius of the galaxy,  $b_n$  is a free parameter which ensures the correct integration properties at  $r_e$ , and  $n$  is the Sérsic index, which describes the brightness concentration curvature of the galaxy.

As summarized in Table 1, we simulated galaxies ranges  $16 \leq \text{mag} \leq 23$  and Sérsic index  $0.3 \leq n \leq 6.2$ . The radius of the galaxies varies in the interval  $1.5 \leq \text{Radius (pixel)} \leq 31.6$  ( $0.09 \leq \text{Radius (arcsec)} \leq 1.9$ ). The axis ratio( $q$ ) of the galaxies ranges within 0.2 and 0.8.

As we will see in the next sections, we used 50 000 simulated galaxies to train and validate our code, and a separate set of other 5000 simulated galaxies to compare the performance of our method with GALFIT.

## 3 INTRODUCTION TO CNNS

Deep artificial neural networks are a particular subclass of artificial neural networks that have allowed a dramatic progress in image and natural language processing in the last years. In this section, we give a very brief introduction to these methods, and we refer to Schmidhuber (2015) for a thorough review of the field.

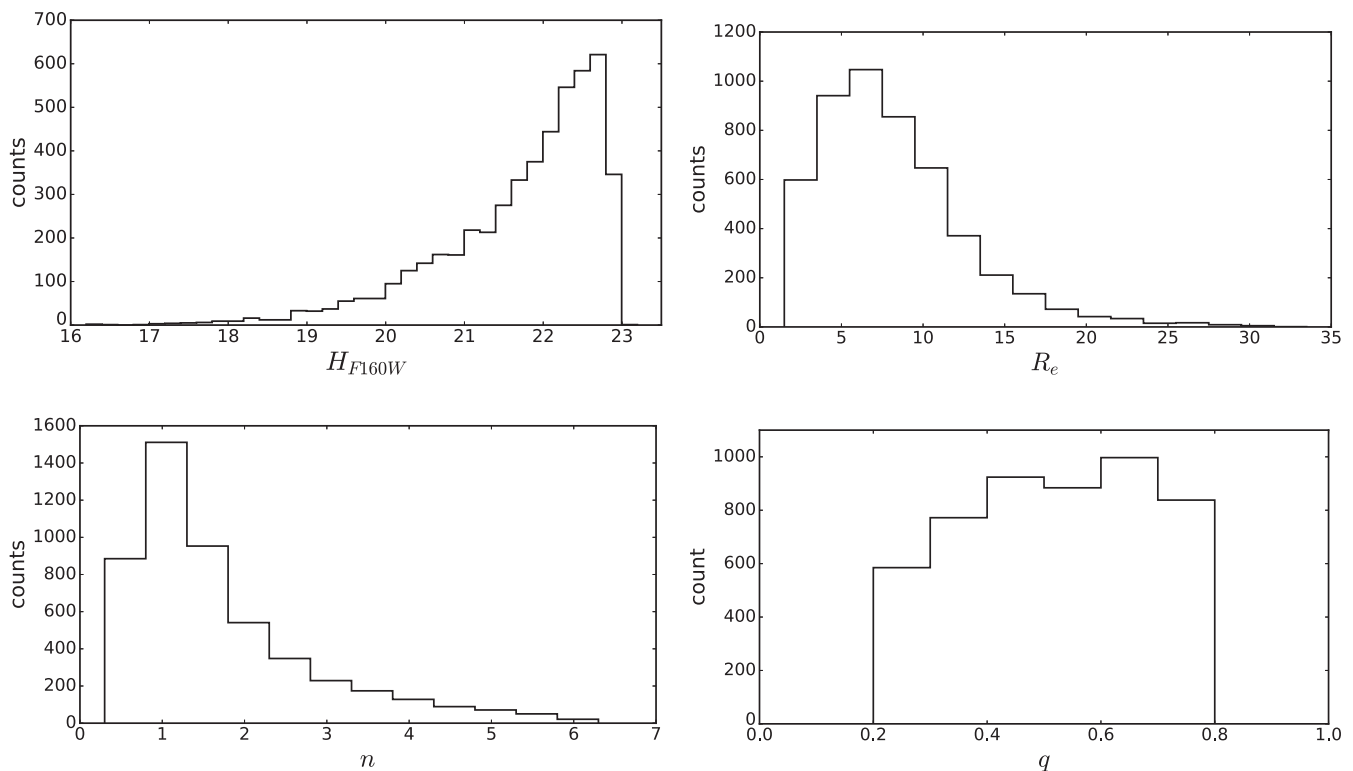
A standard Artificial Neural Network (ANN) consists of many simple, connected processors called neurons, each of them producing a sequence of real-valued activations. Neurons are organized in layers, with the neurons of the input layer directly activated from the features of the input data, and the neurons of the hidden layers activated through weighted connections from previously activated neurons. Analogous to biological systems, where learning involves adjustments in the synaptic connections that exist between neurons, in artificial neural networks the learning process consists in finding the weights that infer the desired output from the features of the input data. Therefore the tensorial output  $\mathbf{h}^l$  of a given layer  $l$  of the network is given by:

$$\mathbf{h}^l = f(\mathbf{W}^l \cdot \mathbf{h}^{l-1} + \mathbf{b}^l), \quad (2)$$

where  $f$  is a non-linear function called *activation function*,  $\mathbf{W}^l$  is the weight matrix, and  $\mathbf{b}^l$  is a biases vector. The biases correspond to the additive constants of the linear combinations that feed each neuron. The network is trained to obtain a specific output for a given input minimizing the *loss function* and thus optimizing the parameters of weights and biases. We refer to Lecun (1989) for their complete description. Briefly, CNNs use convolution layers in place of general matrix multiplication in at least one of the layers, operation usually denoted as  $S(i, j) = (I * K)(i, j)$  with  $K$  being the *kernel* of the convolution function, and the output  $S(i, j)$  known as *feature map*. This results in local connections, with each neuron connected to a sub-sample of the input instead of all the inputs. After the convolution in each neuron follows an *activation function* operation like in ordinary ANNs. Typical components of a CNN include one or more *max-pooling* layers that are subsampling layers where the feature map is downsampled. The max-pooling operation is usually obtained by applying a max filter to (usually) non-overlapping subregions of the initial representation that reduces the output dimensionality while keeping the most salient information. Other typical elements are the *dropout layers* (Srivastava et al. 2014) which are a regularization technique introduced to reduce overfitting. The *fully connected* layer is the final layer of a CNN, where each neuron is completely connected to the other neurons. In a CNN, each layer applies different filters, and each one of these filters detects a specific feature of the inputs. For example, when the inputs are images, different filters can learn from the raw pixels the edges of the images, then other filters can detect simple shapes of the images, which are then used to detect higher level features. The last fully connected layer may then be a classical ANN that uses the recovered information for classification or regression tasks. The implementation of the convolution greatly improves a machine learning system for processing grid-like topology data, like 2D grids of pixels.

## 4 METHOD DESCRIPTION: DEEPLGATO

Our surface brightness fitting code was developed as a CNN used for regression tasks, that in our problem consists in the prediction of galaxy structural parameters given their 2D fits images as input. We called it *DeepLeGATo*, standing for Deep Learning Galaxy Analysis Tool. We implemented our code using *Keras* (Chollet 2015) on top of *Theano* (Bastien et al. 2012), two frameworks commonly used to build Deep Learning models. Our architectures were inspired by VGG-net (Simonyan & Zisserman 2014), the main difference lies in the use of a noise layer. We trained and tested many different CNN architectures and we got the best results with two of them; the first of them will be described in Section 4.1 and the second in Section 4.2.



**Figure 1.** Distributions in magnitude ( $H_{F160W}$ ), half light radius in units of pixels ( $R_e$ ), Sérsic index ( $n$ ), and axis ratio ( $q$ ) of the 5000 *HST*/CANDELS galaxies used in this work. See Section 2.1

In Section 4.3 we describe how we trained and validated our models, in the next section we discuss in detail the performance of the networks on a sample of 5000 galaxies excluded from the training and validation.

#### 4.1 Architecture 1

Architecture 1 of our model is schematically illustrated in the upper image of Fig 2. First, we apply zero-centred additive Gaussian noise to the input images ( $128 \times 128$  pixels), then they are processed by two 2D convolution layers with a  $4 \times 4$  filter size and, finally, subsampled by a  $2 \times 2$  max pooling and a dropout layer. The Gaussian layer, used both in this architecture and in the other one, adds robustness to the model by decreasing its sensitivity to small changes of the input data. Other three units follow with the same configuration of convolutional and max pooling layers, but with a growing dimensionality of the output space (i.e. the number of filters in the output) and a reducing filter size ( $4 \times 4$ ,  $3 \times 3$ ,  $2 \times 2$ ) in the convolutions. Only the first of these three units is followed by a dropout layer. In this architecture we have therefore a total of eight convolutional layers, four max pooling layers, and two dropout layers. Each one of the convolutional layers is followed by a rectified linear unit (ReLU) step. The output of all these units is then processed through three fully connected layers with the decreasing number of neurons (128, 64, 1).

#### 4.2 Architecture 2

The architecture 2 of our model is illustrated in the bottom image of Fig 2. Considering the architecture 1 as the reference, here the input images ( $128 \times 128$  pixels) are first processed by the block composed of two 2D convolution layers with a  $4 \times 4$  filter size,

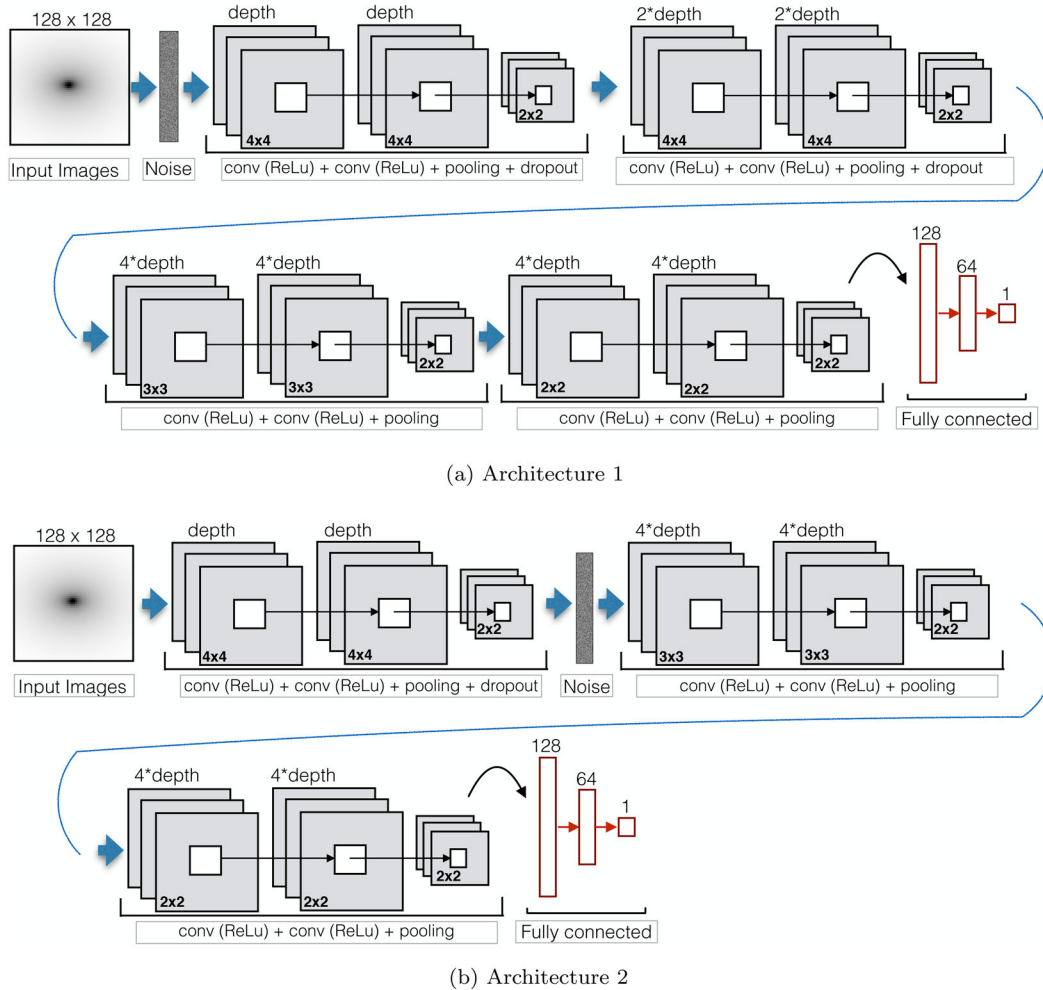
and the  $2 \times 2$  max pooling followed by the dropout layer. After this step, the zero-centred additive Gaussian noise is applied. Other two units follow with the same configuration of convolutional and max pooling layers, but with a growing dimensionality of the output space. Only the first of these three units is followed by a dropout layer. In this architecture we have therefore one block less respect to the architecture 1, for a total of six convolutional layers, three max pooling layers and one dropout layer. Also here, each one of the convolutional layers is followed by an ReLU step, and the block of three fully connected layers is identical to architecture 1.

#### 4.3 Training and validation

Our CNNs have been trained and tested on the simulated data described in Section 2.2. Each parameter is estimated independently, i.e. each CNN is trained and tested using only one parameter at a time as target. The size of the training set was 50k, divided in the proportion of 4/5 for the training and 1/5 for the validation. For the regression problem we used a mean absolute error cost function on a normalized version of variables, i.e. values are in the interval  $[-1, 1]$ . For weights updates we used an adaptive moment estimation (ADAM; Kingma & Ba 2014).

In order to increase the size of the training set and make the model more robust and invariant to specific transformations, we performed *data augmentation* on the training set. Different types of *image augmentation* effectively improve the quality of generalization of CNNs (Krizhevsky et al. 2012). In particular, we applied:

- (i) Random shifts for all parameters, used in order to make the model insensitive to centring. In particular, the images were randomly shifted of 0.05 times the total width of the image.



**Figure 2.** The upper image (a) shows the scheme of Architecture 1 used by *DeepLegato*. The input image is processed adding Gaussian noise, then the first two blocks are composed of two convolution layers with ReLU activation functions, followed by max pooling layers and dropout. The second two units are similar to the previous, but they do not apply dropout. The last block is composed of three connected layers, which outputs the estimation of the evaluated parameter. See Section 4.1 for details. The bottom image (b) shows Architecture 2 used by *DeepLegato*. Here the first block is composed of two convolution layers with ReLU activation function, followed by a max-pooling layer and a dropout. After the addition of a Gaussian noise layer, two units follow with similar architecture as the first one, but they do not apply dropout. The last block is composed of three connected layers, which outputs the estimation of the evaluated parameter. See Section 4.2 for details

(ii) Random horizontal and vertical flips of the images for all parameters.

(iii) Random zoom in the the images within the range [0.7, 1.3] only for the regression of the radius and the Sérsic index.

During the training, we always initialized the weights of our model with random normal values. We *warm up* the training (Huang et al. 2016) of the CNN for 10 epochs, using no exponential decay rate and a starting learning rate of 0.001. During the pre-warming and the subsequent proper fit, we use the ADAM optimizer, which improves and stabilizes the learning rate. After the *warm-up* phase the networks are trained using an *early stopping method*, for a maximum number of 300 epochs. The early stopping method consists in stopping the training if a monitored quantity does not improve for a fixed number (called *patience*) of training epochs. The quantity that we monitored and minimized was the *mean absolute error* of the regression on the parameter of the validation sample, and the patience was fixed to 20.

## 5 TESTS ON SIMULATED DATA

The two CNNs architectures that we have described in the previous section have been chosen from among several different architectures on the basis of their good performance on the validation set. However, in order to finally judge those two architectures, we selected a third test set of simulated galaxies excluded from the training and validation of the models. This way, we evaluate the two models without incurring the risk of meta-training that makes the test set work as a second training set. In Section 5.1 we describe the metric used for this test and the comparison of the two architectures performance. In Section 5.2 we compare the performance of best CNNs with the predictions of GALFIT run on the same data test.

### 5.1 CNNs performance on simulated data

The test set includes 5000 of the simulated galaxies described in Section 2.2. As metric to evaluate the regression, we used the *coefficient of determination*  $R^2$  between the predicted parameter and the ground truth is given from the galaxy simulations. The  $R^2$  is

**Table 2.** The coefficient of determination of the two CNN architectures for the parameters of the light profile fitting, for 5000 simulated galaxies excluded from the training and validation of the methods.

Parameter	$R^2$ simulated data		GALFIT
	Architecture 1	Architecture 2	
Magnitude	0.961	<b>0.997</b>	0.983
Radius	0.899	<b>0.972</b>	0.877
Sérsic index	<b>0.968</b>	0.881	0.607
Axis ratio	<b>0.983</b>	0.959	0.903

a standardized measure of the degree of the regression accuracy defined as:

$$R^2 = 1 - \frac{\sum_i^n (y_i - f_i)^2}{\sum_i^n (y_i - \bar{y})^2}, \quad (3)$$

where  $f_i$  is the predicted value of the true (or input) value of variable  $y_i$ , and  $\bar{y}$  is the mean of the whole set of  $n$  input data. Essentially,  $R^2$  measures how much better we can do in predicting  $y$  by using our model instead of just using the mean as a predictor. When the predicted values come from a least-squares regression line, the coefficient of determination measures the proportion of total variation in the response variable that is explained by the regression line. In fact, in the second term of the above equation, the numerator is the squared sum of the regression errors and the denominator measures the deviations of the observations from their mean. The objective of ordinary least squared regression is to get a line which minimizes the sum squared error. The default line with minimum sum squared error is a horizontal line through the mean. So in the case a least-squares regression line is used to predict the values of  $y$ , then  $0 \leq R^2 \leq 1$ , with  $R^2 = 0$  if the regression line explains no value and  $R^2 = 1$  if it explains the 100 per cent of the variation in the response variable. On the other hand, if we use a different model  $f$  to predict the values of  $y$ , then  $R^2$  naturally ranges between  $-\infty \leq R^2 \leq 1$ , with negative values meaning that a horizontal line at the mean  $\bar{y}$  actually explains the data better than the model  $f$ . The  $R^2$  can be used as an alternative to the *mean squared error*, and the two measurements are related by  $R^2 = 1 - \frac{\text{MSE}}{\sigma_y^2}$ , where MSE is the mean squared error and  $\sigma_y^2$  is the squared standard deviation of the dependent variable.

In Table 2 we show the comparison of the CNNs accuracies on the basis of the  $R^2$  value. It is interesting that a particular network architecture works very well for a particular parameter (e.g. architecture 1 for Sérsic index and axis ratio; architecture 2 for magnitude and half-light radius), while it is not so efficient for other. We conclude that the use of only one architecture for all the parameters does not allow us to obtain the best results since some parameters need more level of abstraction than others.

## 5.2 Comparison with GALFIT

In order to compare the performance of our code with standard algorithms used in the literature for galaxy surface brightness fitting, we used GALFIT on the same set of 5000 simulated galaxies used to test the CNNs architectures. The fitted parameters, listed in Table 1, are the magnitude, half-light radius measured along the major axis, Sérsic index, axis ratio (from which we derive the axis ratio). As PSF image we used the same to generate the simulations and, following a procedure similar to the one used in van der Wel et al. (2012), we used SExtractor to input initial guesses for some of the parameters used in the GALFIT configuration file. In particular,

SExtractor provides initial values for magnitude, half-light radius, and axis ratio. A constraint file is also provided to GALFIT to force it to keep the Sérsic index between 0.2 and 6.3, the effective radius between 0.3 and 130 pixels, the axis ratio between 0.0001 and 1, the magnitude, between  $-3$  and  $+3$  magnitudes from the input value given by the SExtractor magnitude.

The accuracy of GALFIT is measured using the same metrics, i.e. the *coefficient of determination*  $R^2$ , and directly compared with the CNNs performance in Table 2. Additionally, in Fig. 3 we compare the best fit given from the CNN and GALFIT on the simulated data as a function of the apparent  $H$ -band magnitude. In particular, we compute the mean difference between the recovered and input parameters for different magnitude bins. The error bars represent the standard deviation in each bin and reflect thus the uncertainty in the measurements.

For these tests, we run GALFIT on a MacBook Pro running Sierra with 3.1 GHz i7 processors and 16 GB of RAM, and our CNNs on Nvidia Titan X GPU. GALFIT takes approximately 3.5 h to fit 5000 galaxies. Our code, once trained, takes less than 4 s on the GPU and about 200 s on the same CPU machine where we tested GALFIT.

From these comparisons on the simulated data between GALFIT and our CNNs, we show that our models are considerably faster and provide one component galaxy structure decomposition equally or more accurate than GALFIT in the range of magnitude, effective-radius and the Sérsic index used to generate our simulated data.

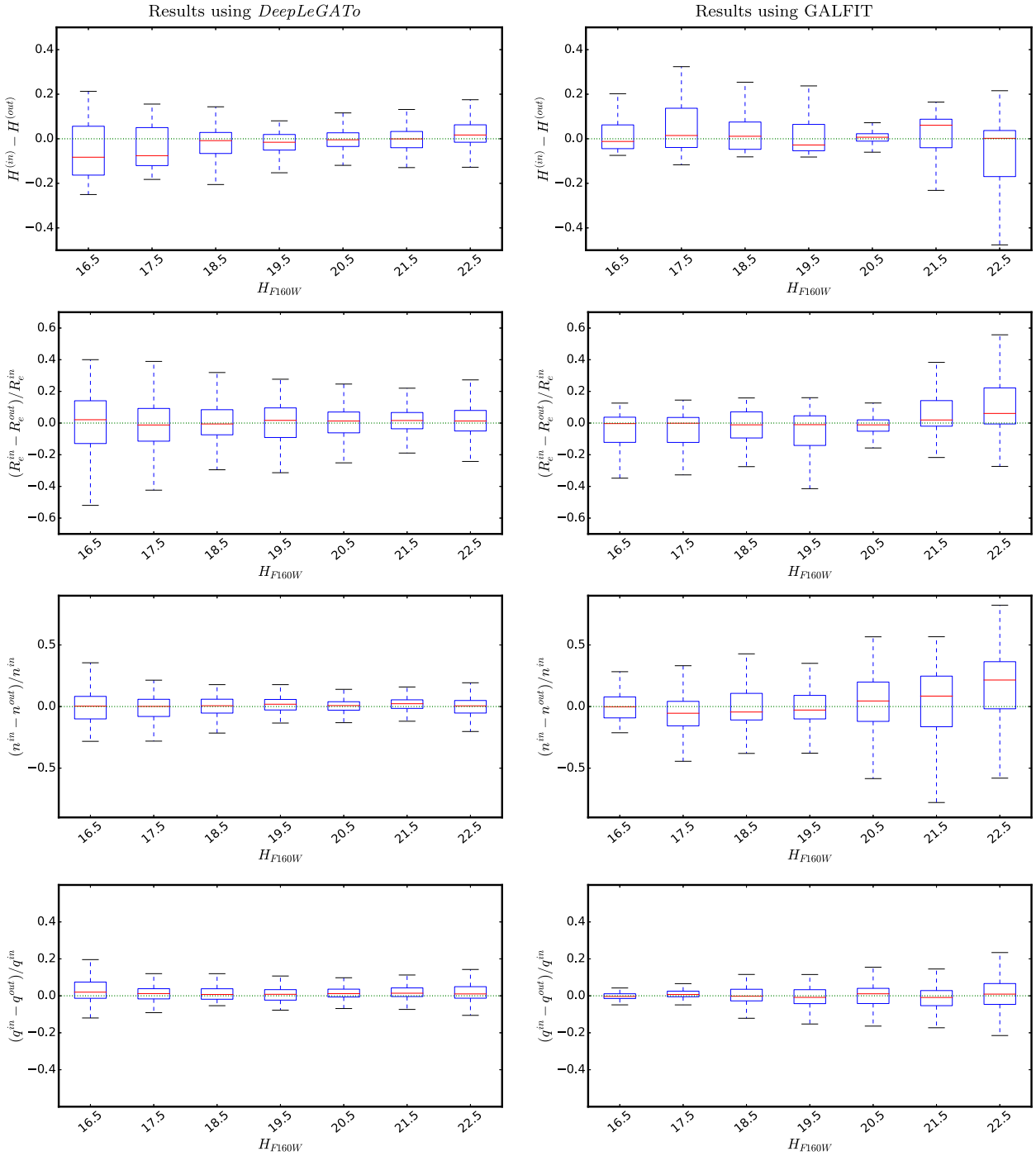
## 6 TESTS ON REAL DATA

Having discussed in the previous section the performance of our code on simulated data, in this section we show its results on the real data presented in Section 2.1.

### 6.1 Direct application of the learned system on real data

Testing on real data is a fundamental step for every profiling method. Galaxy simulations are regular by definition and even when analysing a sample generated with a wide range of structural parameters, we may underestimate the true errors involved in the estimations. Real galaxies may be more asymmetric and difficult to decompose than simulated objects, but above all, the stamps of real galaxies may include the presence of companions that are not included in our simulations. On the other hand, tests on the simulated data presented in the previous section have the great advantage that the *ground truth* is exactly known, i.e. the structural parameters used to generate the data are known by definition. In the case of real data, we do not know the authentic *ground truth*. Testing the accuracy of our code on real galaxies, we need to rely on the comparison with estimations done with some other methods. For this reason, we choose galaxies included in the van der Wel et al. (2012) catalogue, and we use their estimations as *ground truth* for our further tests. This will, of course, add additional scatter into the results since the parameter estimations are certainly affected by both random and systematic (often unknown) errors.

With this assumption, we run our best CNNs architectures (architecture 1 for Sérsic index and axis ratio; architecture 2 for magnitude and half-light radius) on 1000 real galaxies introduced in Section 2.1 and we compare their predictions with those of the van der Wel et al. (2012) catalogue. As in Section 5, we quantify this comparison using the *coefficient of regression*  $R^2$ . When we fit the images of real galaxies with DeepLeGATo trained only on simulations, we do not obtain results as reliable as those obtained on the test sample of



**Figure 3.** Results of profile fitting for the images of 5000 simulated galaxies. The left-hand column shows the results obtained using *DeepLeGATo* (architecture 1 for Sérsic index and axis ratio; architecture 2 for magnitude and half-light radius), the right column the results obtained using *GALFIT*. For each galaxy and for each parameter  $X$  we calculated the difference between the parameter used to simulate the galaxy ( $X^{in}$ ) and the estimation ( $X^{out}$ ) obtained from profile fitting codes. In the first row we show the results for the magnitude  $H_{F160W}$ , in the second for the half-light radius  $R_e$ , in the third for the Sérsic index  $n$  and in the last for the axis ratio  $q$ . The results are shown in bins of magnitude (bin width = 1 mag) and non-parametrically in the form of box plots. As usual, the boxes are delimited by the first and third interquartiles of the data, while the whisker indicates the range between  $\pm 1.5$  the interquartile range (IQR). The red line in the box indicates the median of the data.

simulated data (see Section 5.2). We show this result for the magnitude in panel (a) of Fig. 4, and for all the other parameters in the plots (a) of Figs A1, A2, and A3 of Appendix A. In those figures, on the  $x$ -axis we plot the parameter estimation given from the van der Wel et al. (2012) catalogue, while on the  $y$ -axis we give the

parameter values estimated with *DeepLeGATo*. The same result is given in terms of  $R^2$  for the first column of Table 3.

We qualitatively analysed the reasons of this discrepancy by directly looking at the stamps of the galaxies, both for the simulated data and for the real ones. At first glance, the simulated and real data



**Figure 4.** We show the results of our CNNs in fitting the magnitude of one thousand real galaxies. On the  $x$ -axis we plot the parameter estimation given from the van der Wel et al. (2012) catalogue, while on the  $y$ -axis we give the parameter values estimated with DeepLeGATo. The upper panels show the results obtained for the whole sample before the *domain adaptation* step (panel a) and after the *domain adaptation* (panel b). The three bottom panels show (c) the results obtained for the 142 galaxies whose brightest companion has at least the 50 per cent of their flux; (d) the middle panel shows the results for the 450 galaxies whose companion has less than the 10 per cent of the flux; and (e) the third panel shows the results for the 103 isolated galaxies of our test sample, i.e. without any companion within the stamp

seem to be quite similar, but the differences are evident when we order the data depending on the difference between the parameters predicted by the CNNs and the ground truth. The first two lines of Fig. 5 show 12 randomly selected stamps of simulated galaxies. The third and fourth lines show 12 stamps of real galaxies selected between the ones best predicted by our CNNs (with respect to the estimations given in the van der Wel et al. 2012 catalogue). Finally, the last two lines show 12 stamps of real galaxies selected between the ones worse predicted by our CNNs. Notice that for both best and worse predictions, we chose three galaxies for each of the four parameters. From these images we can see that the CNNs purely trained on simulated data are able to recover the parameters of real galaxies that are as regular and isolated as the simulated. The main issue our method encounters when fitting real data appears to be the presence of bright companions in the stamp, since our simulation

includes only one object per stamp. The CNNs can still give accurate predictions when the real galaxies show smooth asymmetries and in the stamps there are other fainter and smaller galaxy companions. They give unreliable results when the stamps include several brighter companions.

In order to verify the conclusions of this visual analysis, we calculated, for the whole test sample of real data, the number of galaxies having at least one companion close enough to fall within the stamp (i.e. within  $\sim 3''.8$ ). We obtain that almost all the galaxies (897 out of 1000) have at least one companion and that the mean number of companions for galaxy is  $\sim 2.5$ . We verified that, for all parameters, the accuracy of the CNNs predictions is clearly correlated with the presence of a bright companion. In the three bottom panels of Fig. 4 we show this trend for the magnitude. We calculated the ratio between the flux of the fitted galaxy and the

**Table 3.** The coefficient of determination of the light profile fitting obtained with different methods for a sample of 1000 *HST*/CANDELS real galaxies. In the first column the parameters were obtained with *DeepLeGATo* before the domain adaptation step (see Section 6.1). In column 2 we repeat the same test restricting the analysis to 103 *isolated* galaxies, i.e. without neighbour galaxies or stars within the stamp. In column 3 we apply again *DeepLeGATo* to the whole sample of 1000 real galaxies, but after the domain adaptation step (see Section 6.2). Finally, in column 4 we compare the estimations of van der Wel et al. (2012) with those of Dimauro et al. (2017, submitted), i.e. comparing two estimations of two different set-ups of GALFIT (see Section 6.3).

Parameter	$R^2$ Real data			
	Before TL	BTL isolated	After TL	2 GALFIT
Magnitude	0.795	0.979	0.980	0.984
Radius	-0.431	0.630	0.813	0.860
Sérsic index	-0.331	0.516	0.813	0.819
Axis ratio	0.773	0.915	0.934	0.914

one of its brighter companion. In panel (c), we show the results obtained fitting the 142 galaxies whose companion has at least the 50 per cent of their flux. In panel (d) we show the results for the 450 galaxies whose companion has less than the 10 per cent of the flux of the galaxy. Finally, panel (e) shows the results for the 103 isolated galaxies of our test sample, i.e. without companion within the stamp. For the latter sample, we also computed the  $R^2$ , reported in the second column of Table 3. In Figs A1, A2, and A3 of Appendix A, and in the second column of Table 3, we repeat the same analysis for the half-light radius, the Sérsic index, and the axis ratio. We conclude that on isolated galaxies, our machine trained on simulations is able to retrieve accurately the structural parameters.

## 6.2 Domain adaptation

In the previous subsection, we show that a direct application on real data of the CNN models trained on the simulated data does not lead to predictions comparable to the ones obtained using GALFIT if galaxies have bright neighbours. This is because we did not include companions in our simulations. One simple possibility to overcome this problem is to produce more realistic simulations. In this section, we explore an alternative based on domain adaptation between networks. There is a vast literature in computer science dealing with this kind of problems, where methods are trained and evaluated on a certain kind of image distribution but then they are applied to changing visual domains. In general, visual domains could differ in some combination of (often unknown) factors, including viewing angle, resolution, intra-category variation, object location, and pose. Studies have demonstrated a significant degradation in the performance of image methods due to these domain shifts. Therefore, methods of so-called *domain adaptation* have been developed to deal with these situations, where the task of the machine learning method remains the same between each set, but the input distribution is slightly different.

Adopting a *domain adaptation* strategy, we saved our best models trained and validated on simulated data, and we repeated their training and validation using a small sample of real data. The core idea of this strategy is that the same representation learned on simulated data may be useful to adapt the learning system on the second setting of real data. This way, we exploit what the CNNs have learned on one setting to improve the generalization in another one.

In Section 2.1 we described a sample of 5000 real galaxies included in the van der Wel et al. (2012) catalogue. We divided that

sample into a test sample *A* of 1000 galaxies, and a training sample *B* including the remaining 4000 galaxies. We trained and validate our CNNs with different subsets ( $B_1, B_2, B_3, \dots$ ) of *B*, having size varying between 50 and 4000. In agreement with the procedure that we followed in this work, each of the  $B_i$  samples was divided in the proportion of 4/5 for the training and 1/5 for the validation. After the training and validation of the *domain adapted* CNN, we applied the result of the new model on the sample *A* and we calculated  $R^2$  of the prediction for each galaxy structural parameter retrieved. Through all this step we considered the structural parameters estimations given in the van der Wel et al. (2012) catalogue as the *ground truth* of our models.

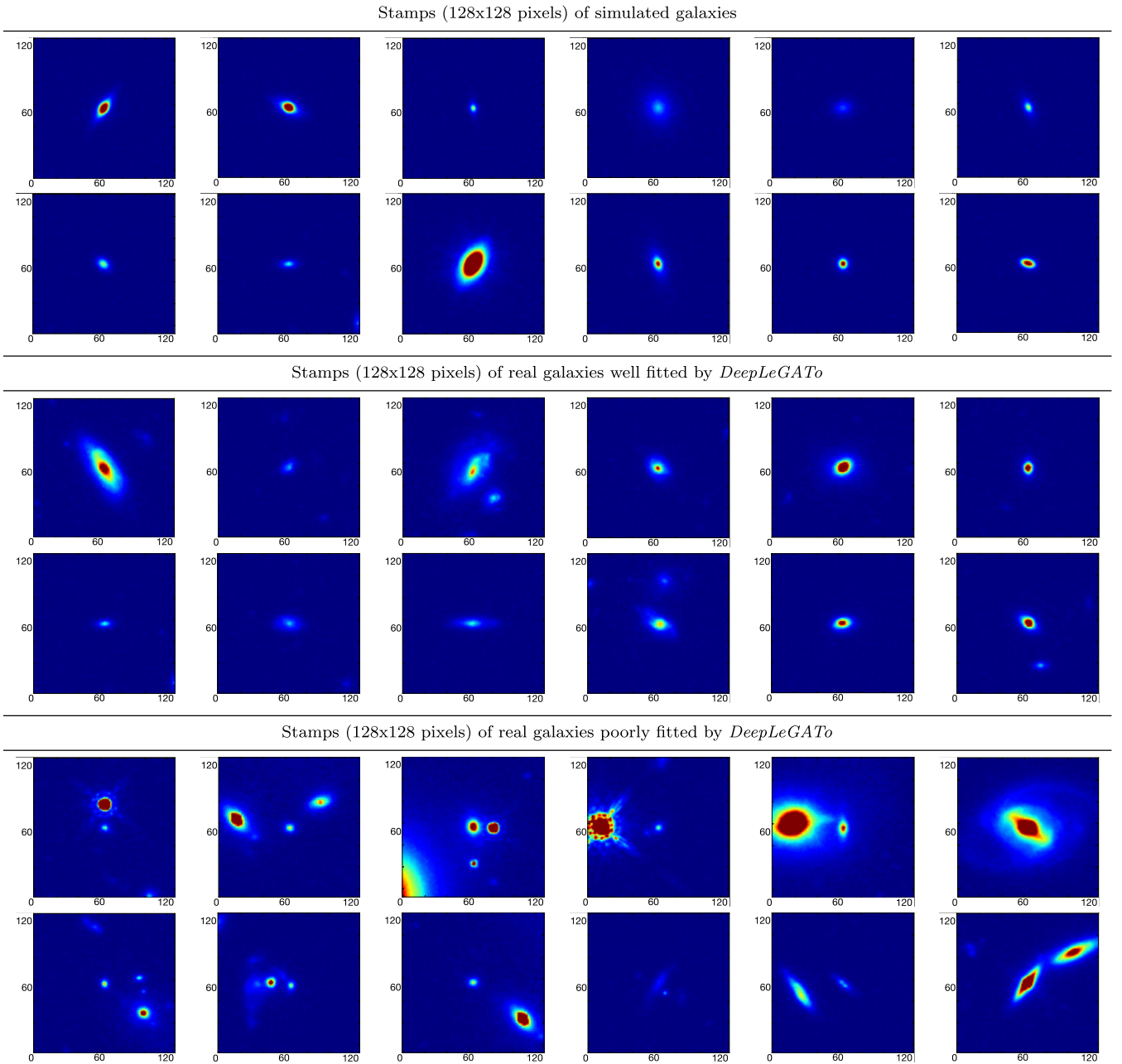
In Fig. 6 we show how  $R^2$  of the prediction varies as a function of the size of the real data used to train and validate the CNN. The blue lines represent the behaviour of our CNNs previously trained on simulated data, to which we then applied *domain adaptation*; the green star at the beginning of the blue lines represents the  $R^2$  of the prediction before the *domain adaptation*. The red lines represent the behaviour of our CNNs directly trained on the  $B_i$  samples of real data (i.e. without *domain adaptation*). Each parameter is estimated independently by the CNN that showed the best accuracy for the profiling of the simulated data (see Table 2). For the magnitude, with only 50 real galaxies used as *domain adaptation*, we obtain an  $R^2 \sim 0.9$  and with 200 real galaxies an  $R^2 \sim 0.97$  (i.e. a 0.4 per cent of the 50k simulated galaxies formerly used to train and validate the CNN). In the case of the magnitude, without *domain adaptation* (red line), with 800 real galaxies we already reach an  $R^2 \sim 0.96$ . For the half-light radius, the  $R^2$  has a value of  $\sim -0.43$  when trained only with simulated data (the green star), and it improves to  $\sim 0.62$  with *domain adaptation* made with only 50 real galaxies. It reaches an  $R^2 \sim 0.81$  with 600 real galaxies, while without *domain adaptation*, it needs  $\sim 1500$  galaxies to reach an  $R^2 \sim 0.75$ . In the case of the Sérsic index, the best  $R^2 \sim 0.79$  is obtained with *domain adaptation* made with 400 real galaxies, while the best performance for the CNN trained without *domain adaptation* is  $R^2 \sim 0.76$  obtained with 1000 real galaxies. The trend for the axis ratio is smoother, with the blue line (*domain adaptation curve*) improving in the whole range of sizes and obtaining  $R^2 \sim 0.93$  when trained/validated with 4000 real galaxies. On the contrary, the red line (trend without *domain adaptation*) reaches a plateau of  $R^2 \sim 0.70$  with 800 real galaxies. We notice that for all parameters the blue line is always above the red line, i.e. when using *domain adaptation curve* the CNNs need fewer real galaxies examples to reach their best performance on real galaxies and they are always more accurate than CNNs not previously trained with simulated data.

In Fig. 4 and in Appendix A we compare the result of the CNNs profiling of the one thousand galaxies before and after the *domain adaptation*.

## 6.3 Comparison with GALFIT

The results of the GALFIT predictions greatly depend on the initial parametrization used. Different parametrization may easily lead to different results. To take this effect into account, in the third column of Table 3 we compare, for our test set of real data, the predictions of van der Wel et al. (2012) catalogue (used as ground truth) with the ones of another catalogue using GALFIT (Dimauro et al. in preparation). As we can see,  $R^2$  of our models after the *domain adaptation* are very similar to the results obtained for the same set of data from these two catalogues using GALFIT.

In Fig. 7 we detail the comparison between our CNNs predictions and the two GALFIT catalogues, as a function of the magnitude.

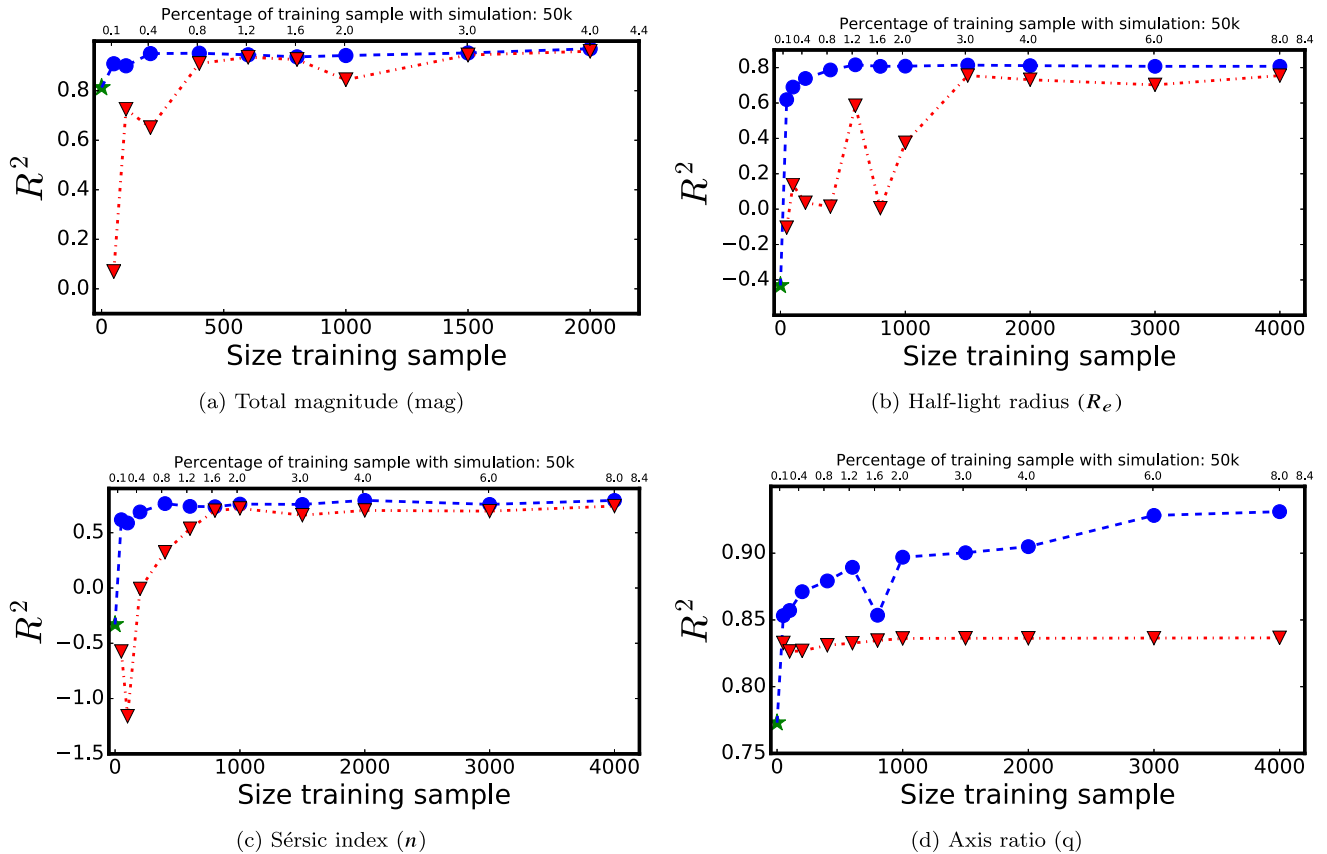


**Figure 5.** The first two lines we show 12 randomly selected stamps of simulated galaxies. The third and fourth lines show 12 stamps of real galaxies selected between the ones better predicted by our CNNs. Finally, the last two lines show 12 stamps of real galaxies selected between the ones worse predicted by our CNNs. From these images we can see that the CNNs purely trained on simulated data are able to recover the parameters of real galaxies that are as regular and isolated as the simulated. The CNNs can still give accurate predictions when the real galaxies show smooth asymmetries and in the stamps there are other fainter and smaller galaxy companions. They give worse results when the stamps include several brighter and bigger galaxy companions.

## 7 CONCLUSIONS AND PERSPECTIVES

In this work, we present *DeepLeGATo*, a CNN method designed for 2-D light profile fitting of one-component galaxies. We trained, validated, and tested *DeepLeGATo* on an extensive set of simulations of *HST*/CANDELS galaxies images and we find it to be robust in terms of parameter recoverability and consistent with the results obtained with *GALFIT*. We find that the use of different architectures depending on the output parameter helps to improve the performance of the models, therefore *DeepLeGATo* uses two different architectures. On simulated data, our method obtains more

accurate results than *GALFIT* for the parameters (magnitude, half-light radius, Sérsic index, and axis ratio) of the structural decomposition. Moreover, it is  $\sim 3000$  times faster when running on GPU, and  $\sim 50$  times when run on the same CPU machine. On real data, *DeepLeGATo* trained on simulations behaves similarly to *GALFIT* on isolated and fairly regular galaxies. Moreover, after a *domain adaptation* step made with the 0.1–0.8 per cent of the size of the training set, *DeepLeGATo* obtains results consistent with the ones presented in the van der Wel et al. (2012) catalogue even for galaxies having several bright companions. Considering this test, we



**Figure 6.** We show  $R^2$  for the profile fitting of 1000 real CANDELS/*HST* galaxies, obtained using our CNN code. The values of  $R^2$  are plotted as a function of the size of real-data sample used to train and validate the method. In particular, the blue line represents the behaviour of our CNN previously trained on simulated data and then *domain adapted* using real galaxies. The starting point of the blue line, the green star, represents the  $R^2$  value obtained using the CNN purely trained on simulated data (see Section 6.1). The red line represents the behaviour of our CNN directly trained on real data (i.e. without *domain adaptation*). The points on the x-axis, i.e. the size of the training/validation sample, are 50, 100, 200, 400, 600, 800, 1000, 1500, 2000, 3000, and 4000. See Section 6.2.

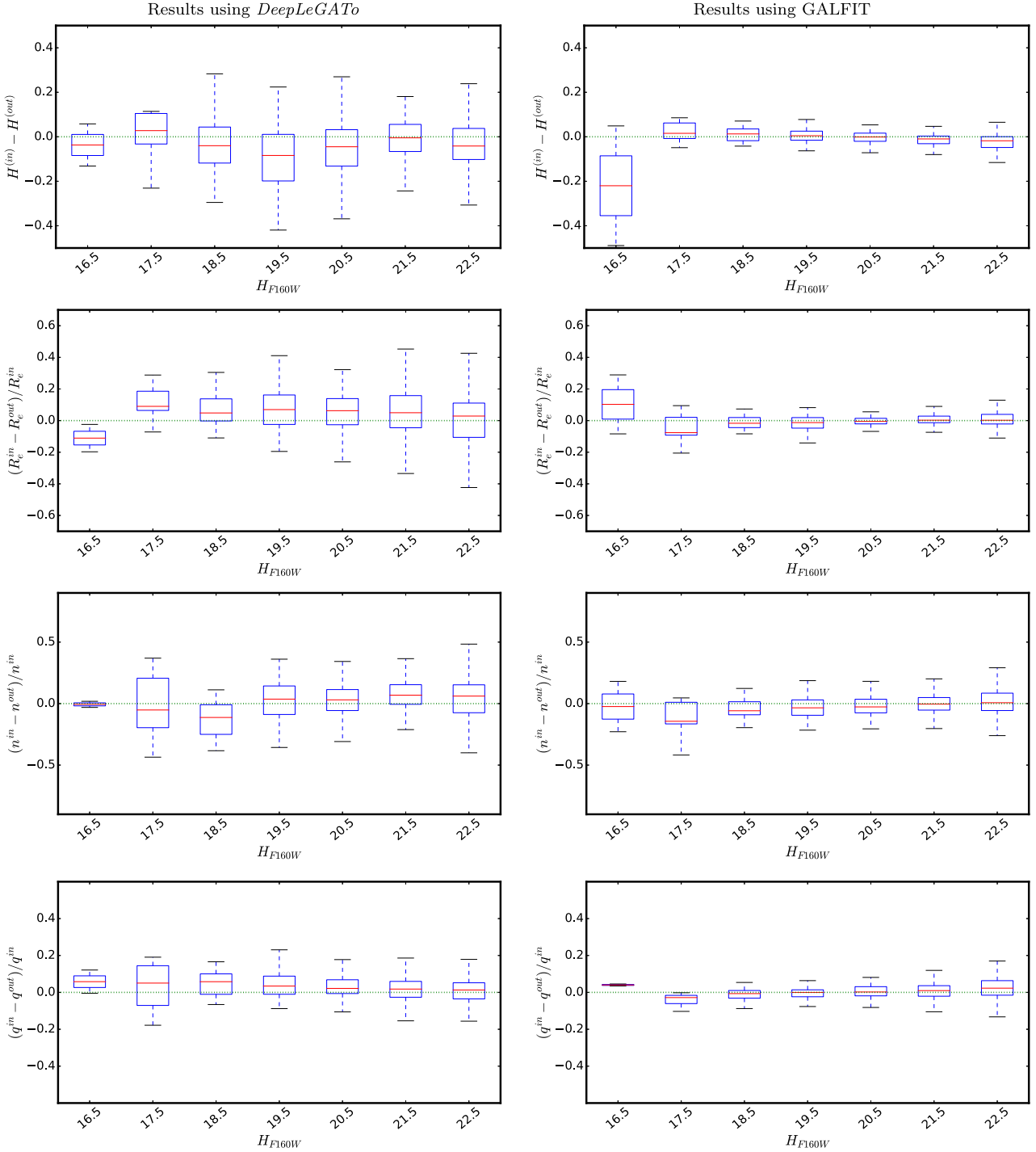
conclude that our method is able to obtain reliable results either using more complex and realistic simulations for its training, either with a *domain adaptation* step made with a sample of reliable estimations.

Overall, in this work we prove that deep neural networks represent an exciting prospect for conducting large-scale galaxy-decomposition, as they are capable of automated feature extraction and do not need a hand-made user-defined parameter set-up. While the accuracy of other methods greatly depends on the choice of the input set-up of parameters, initial conditions, and centring. Deep learning and CNNs, in particular, have the potential to significantly cut down on the need for human visual inspection and make the galaxy decomposition a powerful tool in the era of new and future wide-field surveys such as LSST, Euclid, and WFIRST.

*DeepLeGATo* aims to be a first step towards the systematic use of deep learning methods for fast, accurate, and precise measurements of galaxy structural parameters. Future steps are needed before this method can be implemented effectively on large photometric data sets. A fundamental issue that we plan to address is the measurement of uncertainties. Realistic error bars are crucial to confirm or rule out models when compared to data. Several solutions to this problem may be viable outside the CNN framework. For instance, calculating the residuals after subtracting away from the galaxy image a model generated with the fitted parameters. More elegant solutions having

interest beyond our specific task would implement the estimation of the uncertainties within the CNN framework although this is still an unsolved issue. We tested one possible approach, consisting in fitting several slightly different images of the same galaxy. We test the self-consistency of the CNN regression using different observations (or artificially disturbed images) of the galaxies, thus obtaining an empirical sample of the predictive distribution for each one of them. From the latter we infer an empirical estimator for the predictive variance (our uncertainty in the estimations). Another possible solution involves the use of Bayesian neural networks (BNNs), that are currently considered the state of the art for estimating predictive uncertainty (Gal 2016). BNNs learn a-posterior distribution over the parameters of the neural network and use this approximate distribution as a prior to provide a complete probability distribution of the estimations. Classical deep neural networks provide only a single estimation.

As shown in the paper, in this initial test we excluded all extreme cases from our analysis, including very round or flat galaxies. Future and ongoing work includes the extension of the parameter space, and the application of CNNs on more complex cases consisting of two-component bulge-disc galaxies, implementing a variable PSF and making the algorithm more robust for dense and noisy regions of the sky. Last, we plan to render *DeepLeGATo* freely available with a full set of instructions to adapt the method to different surveys and galaxy samples.



**Figure 7.** Results of profile fitting for the images of 1000 real *HST*/CANDELS galaxies. The left-hand column shows the results obtained using *DeepLeGATo* after domain adaptation, and the right-hand column the results obtained using two different catalogues based on the use of *GALFIT*. For each galaxy and for each parameter  $X$ , we calculated the difference between the parameter estimation given in Van der Wel et al. (2012) ( $X^{in}$ ) and, on the left-hand column the estimation ( $X^{out}$ ) obtained from *DeepLeGATo*, and on the right the estimation published in Dimauro et al. (2017, submitted). The results are shown in bins of magnitude (bin width = 1 mag) and non-parametrically in the form of box plots. As usual, the boxes are delimited by the first and third interquartiles of the data, while the whisker indicates the range between  $\pm 1.5$  the interquartile range (IQR). The red line in the box indicates the median of the data.

## ACKNOWLEDGEMENTS

This work is funded by the project GAMOCLASS (P.I. MHC) as part as the *appel structuration de la recherche lancé par PSL au titre de l'année 2015*. The authors are also thankful to Google for the unrestricted gift given at the University of Santa Cruz for the project

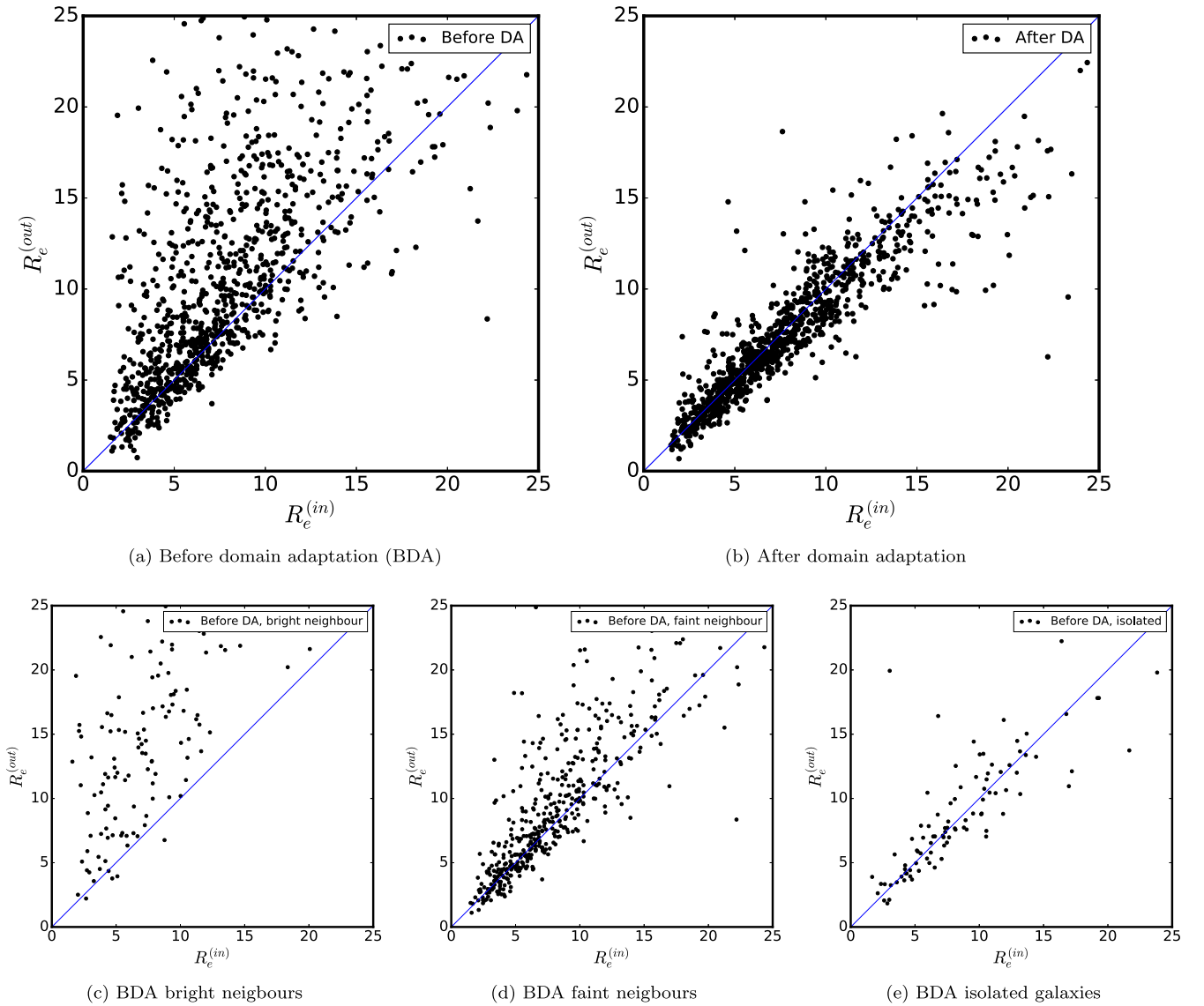
*Deep learning for galaxies*, which contributed to improve this work. MHC would like to thank Sandy Faber, David Koo, and Joel Primack for very useful discussions. The authors gratefully thank the referee, Arjen van der Wel, for the constructive and positive comments which help to improve the paper.

## REFERENCES

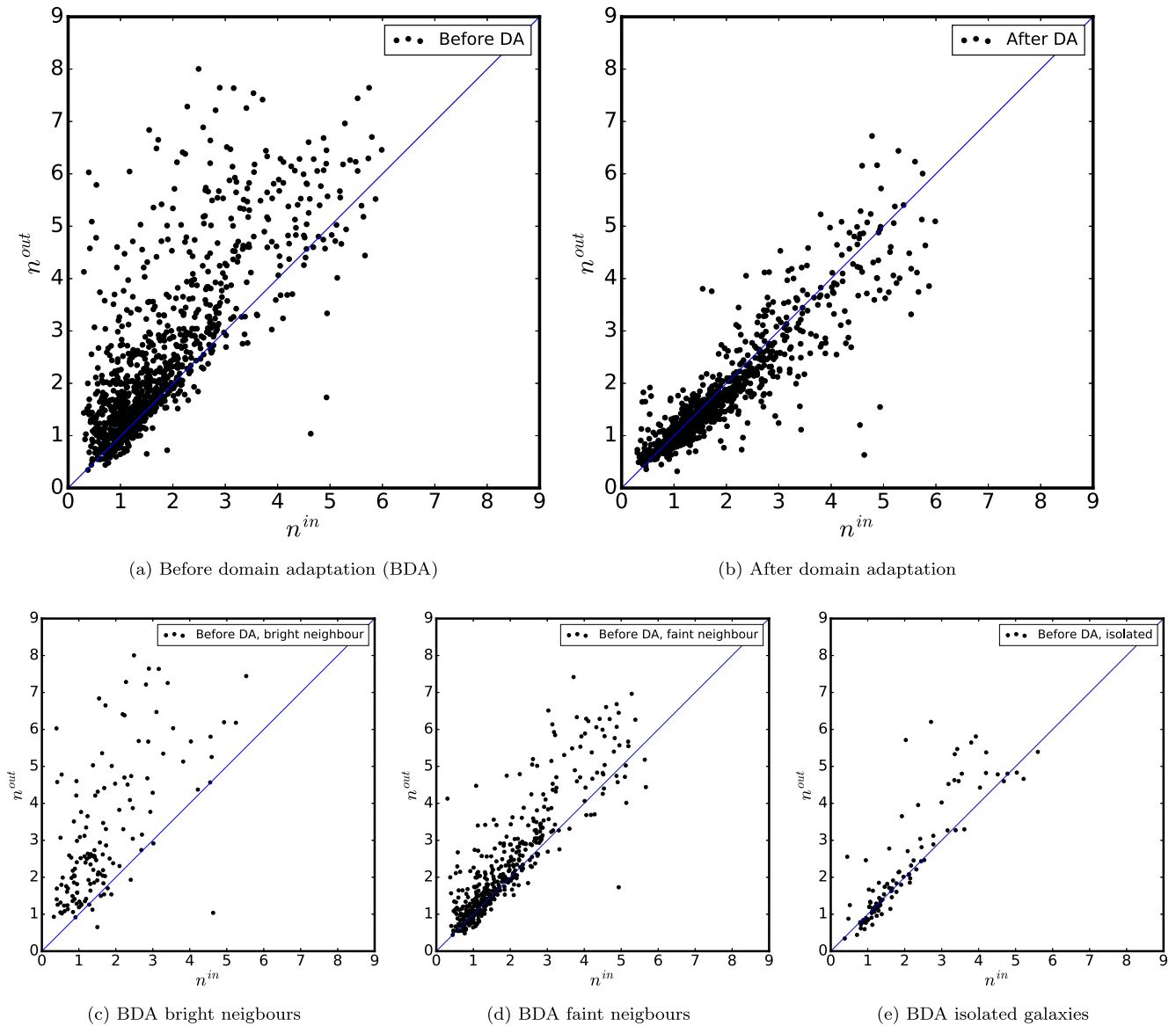
- Barden M., Häußler B., Peng C. Y., McIntosh D. H., Guo Y., 2012, MNRAS, 422, 449
- Bastien F. et al., 2012, preprint ([arXiv:1211.5590](https://arxiv.org/abs/1211.5590))
- Bernardi M., Meert A., Sheth R. K., Vikram V., Huertas-Company M., Mei S., Shankar F., 2013, MNRAS, 436, 697
- Bertin E., Arnouts S., 1996, A&AS, 117, 393
- Chollet F., 2015, keras. Available at: <https://github.com/fchollet/keras>
- de Jong J. T. A., Verdoes Kleijn G. A., Kuijken K. H., Valentijn E. A., 2013, Exp. Astron., 35, 25
- de Vaucouleurs G., 1958, ApJ, 128, 465
- Dieleman S., Willett K. W., Dambre J., 2015, MNRAS, 450, 1441
- Dimairo P. et al., 2017, MNRAS, submitted
- Driver S. P., Allen P. D., Liske J., Graham A. W., 2007, ApJ, 657, L85
- Driver S. P. et al., 2009, Astron Geophys, 50, 5.12
- Frontera-Pons J., Sureau F., Bobin J., Le Floch E., 2017, A&A, 603, A60
- Gal Y., 2016, PhD thesis, Univ. Cambridge, Cambridge
- Grogin N. A. et al., 2011, ApJS, 197, 35
- Huang G., Liu Z., Weinberger K. Q., van der Maaten L., 2016, preprint ([arXiv:1608.06993](https://arxiv.org/abs/1608.06993))
- Huertas-Company M. et al., 2015, ApJS, 221, 8
- Kelvin L. S. et al., 2014, MNRAS, 439, 1245
- Kennedy R. et al., 2016, MNRAS, 460, 3458
- Kent S. M., 1985, ApJS, 59, 115
- Kim E. J., Brunner R. J., 2017, MNRAS, 464, 4463
- Kingma D. P., Ba J., 2014, preprint ([arXiv:1412.6980](https://arxiv.org/abs/1412.6980))
- Koekemoer A. M. et al., 2011, ApJS, 197, 36
- Kormendy J., 1977, ApJ, 217, 406
- Krizhevsky A., Sutskever I., Hinton G. E., 2012, in Pereira F., Burges C. J. C., Bottou L., Weinberger K. Q., eds, Advances in Neural Information Processing Systems. Curran Associates, Inc.
- Lange R. et al., 2016, MNRAS, 462, 1470
- Lecun Y., 1989, in Pfeifer R., Schreter Z., Fogelman F., Steels L., eds, Generalization and Network Design Strategies. Elsevier
- LeCun Y., Bengio Y., Hinton G., 2015, Nature, 521, 436
- Lintott C. J. et al., 2008, MNRAS, 389, 1179
- LSST Science Collaboration et al., 2009, preprint ([arXiv:0912.0201](https://arxiv.org/abs/0912.0201))
- Peng C. Y., Ho L. C., Impey C. D., Rix H.-W., 2002, AJ, 124, 266
- Robotham A. S. G., Taranu D. S., Tobar R., Moffett A., Driver S. P., 2017, MNRAS, 466, 1513
- Rowe B. T. P. et al., 2015, Astron. Comput., 10, 121
- Schmidhuber J., 2015, preprint ([arXiv:1404.7828](https://arxiv.org/abs/1404.7828))
- Sersic J. L., 1968, Atlas de Galaxias Australes. Observatorio Astronomico, Cordoba, Argentina
- Shen S., Mo H. J., White S. D. M., Blanton M. R., Kauffmann G., Voges W., Brinkmann J., Csabai I., 2003, MNRAS, 343, 978
- Simard L. et al., 2002, ApJ Suppl. Ser., 142, 1
- Simonyan K., Zisserman A., 2014, preprint ([arXiv:1409.1556](https://arxiv.org/abs/1409.1556))
- Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R., 2014, J. Mach. Learn. Res., 15, 1929
- van der Wel A. et al., 2012, ApJS, 203, 24
- van der Wel A. et al., 2014, ApJ, 788, 28
- Vikram V., Wadadekar Y., Kembhavi A. K., Vijayagovindan G. V., 2010, MNRAS, 409, 1379
- Wang K., Guo P., Luo A.-L., 2017, MNRAS, 465, 4311
- York D. G. et al., 2000, AJ, 120, 1579

## APPENDIX A: CNNs PREDICTION ON REAL DATA

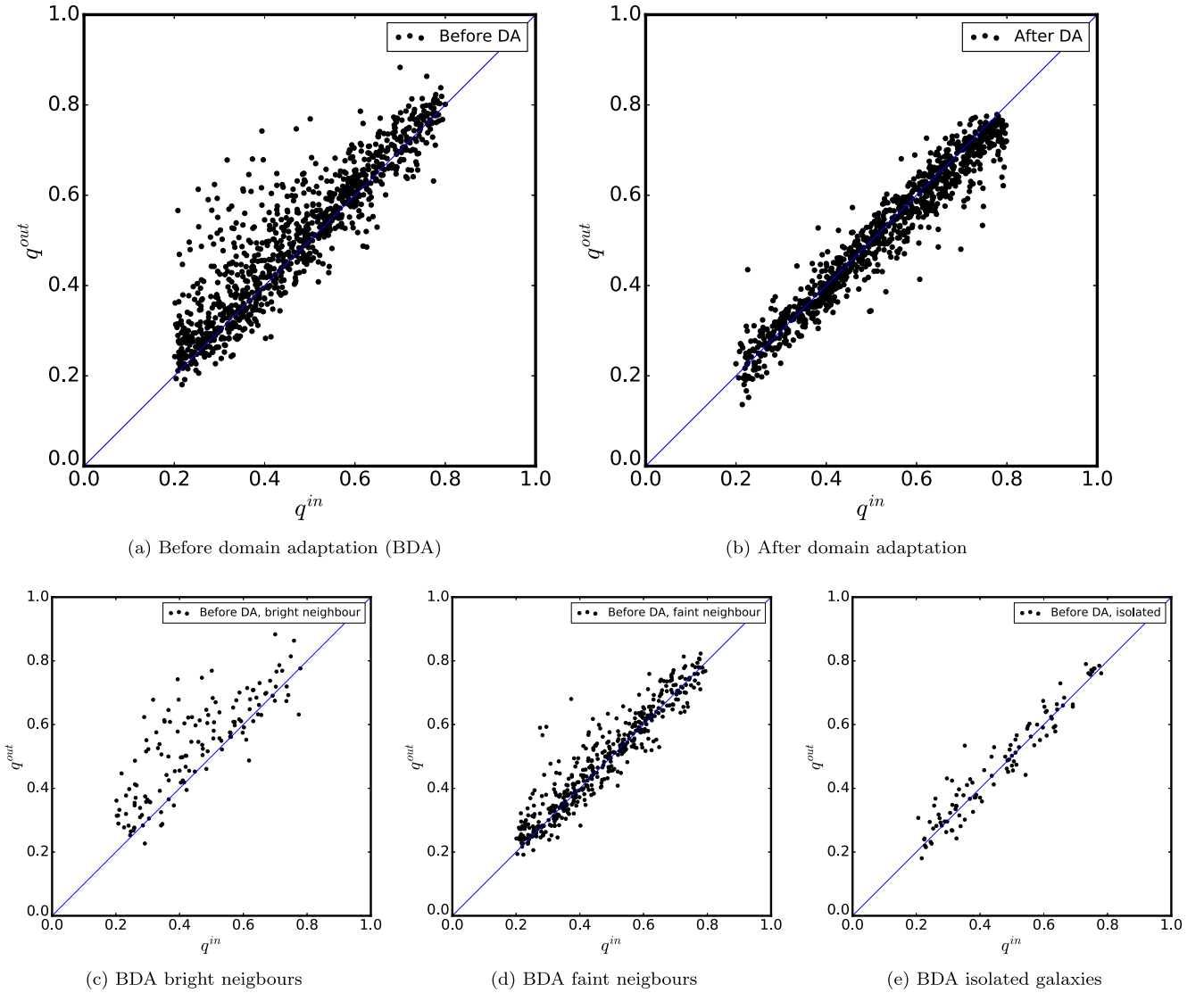
In this appendix we complete the analysis presented in Section 6.1, and we show the results of the application of *DeepLeGATo* to a test sample of real galaxies. In particular, we show the one-to-one scatter-plot comparison of our estimations with those of the van der Wel et al. (2012) catalogue before and after *domain adaptation*. The sample fitted before *domain adaptation* is presented in Figs A1, A2, and A3.



**Figure A1.** We show the results of our CNNs in fitting the half-light radius of one thousand real galaxies. On the  $x$ -axis we plot the parameter estimation given from the van der Wel et al. (2012) catalogue, while on the  $y$ -axis we give the parameter values estimated with DeepLeGATo. The upper panels show the results obtained for the whole sample before *domain adaptation* step (a), and after the *domain adaptation* (b). The three bottom panels show: (c) the results obtained on the 142 galaxies whose companion has at least 50 per cent of their flux; (d) the results for the 450 galaxies whose companion has less than 10 per cent of the flux of the galaxy; (e) results for the 103 isolated galaxies of our test sample, i.e. without companion within the stamp.



**Figure A2.** We show the results of our CNNs in fitting the Sérsic index of one thousand real galaxies. On the  $x$ -axis we plot the parameter estimation given from the van der Wel et al. (2012) catalogue, while on the  $y$ -axis we give the parameter values estimated with DeepLeGATo. The upper panels show the results obtained for the whole sample before *domain adaptation* step (a) and after the *domain adaptation* (b). The three bottom panels show (c) the results obtained on the 142 galaxies whose companion has at least 50 per cent of their flux; (d) the results for the 450 galaxies whose companion have less than the 10 per cent of the flux of the galaxy; (e) the results for the 103 isolated galaxies of our test sample, i.e. without companion within the stamp.



**Figure A3.** We show the results of our CNNs in fitting the axis ratio of one thousand real galaxies. On the  $x$ -axis we plot the parameter estimation given from the van der Wel et al. (2012) catalogue, while on the  $y$ -axis we give the parameter values estimated with DeepLeGATo. The upper panels show the results obtained for the whole sample before *domain adaptation* step (a) and after the *domain adaptation* (b). The three bottom panels show: (c) the first from the left, the results obtained on the 142 galaxies whose companion has at least 50 per cent of their flux; (d) the results for the 450 galaxies whose companion have less than 10 per cent of the flux of the galaxy; and (e) the results for the 103 isolated galaxies of our test sample, i.e. without companion within the stamp.

This paper has been typeset from a  $\text{\LaTeX}$  file prepared by the author.