Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

MNRAS **483,** 2968–2982 (2019)
Advance Access publication 2018 November 29

doi:10.1093/mnras/sty3232

Identification of low surface brightness tidal features in galaxies using convolutional neural networks

Mike Walmsley [1,2★] Annette M. N. Ferguson,[3] Robert G. Mann[3] and Chris J. Lintott [2]

[1] *School of Physics and Astronomy, James Clark Maxwell Building, Peter Guthrie Tait Road, Edinburgh EH9 3FD, UK*
[2] *Oxford Astrophysics, The Denys Wilkinson Building, Oxford OX1 3RH, UK*
[3] *Institute for Astronomy, Royal Observatory Edinburgh, University of Edinburgh, Blackford Hill, Edinburgh EH9 3HJ, UK*

Accepted 2018 November 26. Received 2018 November 26; in original form 2018 September 17

ABSTRACT

Faint tidal features around galaxies record their merger and interaction histories over cosmic time. Due to their low surface brightnesses and complex morphologies, existing automated methods struggle to detect such features and most work to date has heavily relied on visual inspection. This presents a major obstacle to quantitative study of tidal debris features in large statistical samples, and hence the ability to be able to use these features to advance understanding of the galaxy population as a whole. This paper uses convolutional neural networks (CNNs) with dropout and augmentation to identify galaxies in the CFHTLS-Wide Survey that have faint tidal features. Evaluating the performance of the CNNs against previously published expert visual classifications, we find that our method achieves high (76 per cent) completeness and low (20 per cent) contamination, and also performs considerably better than other automated methods recently applied in the literature. We argue that CNNs offer a promising approach to effective automatic identification of low surface brightness tidal debris features in and around galaxies. When applied to forthcoming deep wide-field imaging surveys (e.g. LSST, *Euclid*), CNNs have the potential to provide a several order-of-magnitude increase in the sample size of morphologically perturbed galaxies and thereby facilitate a much-anticipated revolution in terms of quantitative low surface brightness science.

Key words: methods: data analysis – methods: statistical – galaxies: evolution – galaxies: interactions – galaxies: structure – galaxies: statistics.

1 INTRODUCTION

Hierarchical models of galaxy formation suggest that present-day galaxies assemble their mass through the repeated aggregation of smaller systems and through the smooth accretion of gas which fuels *in situ* star formation (e.g. White & Frenk 1991; Abadi et al. 2002). While it is generally agreed that the most massive galaxies have acquired almost all of their stars through mergers, the relative contribution of *in situ* star formation and directly accreted stellar mass remains an open question across much of the galaxy mass spectrum (e.g. Rodriguez-Gomez et al. 2016; Lee & Yi 2017; Qu et al. 2017; Fitts et al. 2018). Furthermore, the rates of major (mass ratio $\geq$1:4) and minor merger events, and their role in shaping various galaxy components, are also not yet well understood (e.g. Lotz et al. 2011; Lofthouse et al. 2017; Martin et al. 2017).

Given the intrinsic uncertainties in *ab initio* modelling galaxy formation within a cosmological context (see the discussion in Hopkins et al. 2018), a purely empirical measure of the frequency and nature of galaxy mergers and accretions is highly desirable. It is well established that such events leave long-lasting observational signatures in the form of low surface brightness tidal streams, shells, and perturbations (e.g. Toomre & Toomre 1972; Quinn 1984; Cooper et al. 2010). In galaxy outskirts, where the dynamical time-scales are several gigayears or longer, these features are predicted to be particularly apparent (Johnston, Hernquist & Bolte 1996; Cooper et al. 2013). Indeed, much stellar substructure of this nature has already been detected in the peripheral regions of the Milky Way (e.g. Belokurov et al. 2006), M31 (see the review by Ferguson & Mackey 2016), and other nearby galaxies (e.g. Martínez-Delgado et al. 2010; Duc et al. 2015). Low surface brightness tidal features are therefore a powerful means to identify systems which have undergone recent mergers and accretions. The morphology and properties of these features hold vital clues to the nature of the events which have created them (Hendel & Johnston 2015; Pop et al. 2017).

★ E-mail: mike.walmsley@physics.ox.ac.uk

One of the main obstacles in such studies is the difficulty in reliably identifying faint tidal features. Part of this problem stems from the fact that morphological merger signatures only persist for a finite duration after an interaction has taken place, with the exact time-scale dependent on the details of the orbital interaction as well as the properties of the host galaxy (Lotz et al. 2008a). Although predicted to be very common, minor mergers are particularly challenging to investigate because they generate faint signatures which are detectable over shorter time-scales (Lotz et al. 2011). Indirect evidence for minor mergers from resulting morphological transformations (e.g. bulge growth) can provide sensitivity to events that have occurred over longer time-scales but it is often difficult to distinguish these transformations from secular processes (e.g. Conselice 2003; Kormendy & Kennicutt 2004; Hopkins et al. 2009).

Another major challenge comes from the process of actually identifying the tidal features on deep galaxy images. Most work to date has focused on visual inspection of individual galaxies or relatively small samples (e.g. Malin & Carter 1983; Martínez-Delgado et al. 2010; Sheen et al. 2012), often on images that have been specifically manipulated in order to enhance the appearance of low surface brightness features (e.g. Miskolczi, Bomans & Dettmar 2011; Hood et al. 2018; Kado-Fong et al. 2018; Morales et al. 2018). However, many important questions about the role of interactions and mergers in driving galaxy evolution require large statistical samples (i.e. several thousand systems or more) for which expert human classification becomes impractical. Unfortunately, there has been relatively little effort to date in devising automatic methods to detect and characterize low surface brightness emission in galaxies and the methods invoked are not particularly well suited to detecting the faint tidal features typical of minor mergers. Techniques may be broadly grouped into two categories – those which rely on model subtraction and those which appeal to non-parametric feature extraction.

Model subtraction methods work by removing the expected flux using a parametric light profile and then quantifying the amount of residual light (e.g. van Dokkum 2005; Tal et al. 2009; Adams et al. 2012). This approach works best on galaxies with smooth radially symmetric morphologies because of the difficulty in constructing light profiles that accurately model the basic morphology. Non-parametric feature extraction methods measure one or several hand-crafted image parameters thought to correlate with post-merger disruption (e.g. Conselice 2003; Lotz, Primack & Madau 2004; Freeman et al. 2013; Pawlik et al. 2016), and then apply selection cuts or machine learning estimation to identify the most likely candidates. These methods allow for a broader range of morphologies to be classified but can be easily confused by complex features such as spiral arms (e.g. Kartaltepe et al. 2010) and are typically only sensitive to certain major merger stages (Lotz et al. 2008b, 2011; Snyder et al. 2015).

Motivated by the desire to develop a more generalized method of tidal debris detection and classification, and one which can be applied to the specific problem of identifying (and ultimately characterizing) faint features around galaxies in large statistical samples, we explore a new approach based on convolutional neural networks (hereafter CNNs). Various authors (e.g. Dieleman, Willett & Dambre 2015; Sánchez et al. 2017) have demonstrated that CNNs can accurately classify general galaxy morphology and recently Ackermann et al. (2018) showed that CNNs can be used to identify merging galaxy pairs in SDSS Data Release 7 images (Darg et al. 2010). Here, we use CNNs with dropout and augmentation to identify galaxies in the CFHTLS-Wide Survey that have faint tidal features in their outer regions. Through application to a galaxy sample that has been previously visually searched for debris features, we demonstrate the reliability and effectiveness of our automated technique. We also show that its performance compares favourably to two other methods that have been recently applied in the literature.

The paper is organized as follows. In Section 2, we describe the sample of galaxies under study. In Section 3, we describe the motivation for our approach and the design, training and performance of a single network, while in Section 4 we discuss the improvements obtained through using an ensemble of several networks. In Section 5, we compare with the current approaches of WND-CHARM (Shamir 2012) and shape asymmetry (Pawlik et al. 2016), and in Sections 6 and 7 we discuss our results and conclusions.

2 DATA

We base our analysis on data products from the Wide component of the Canada–France–Hawaii Telescope Legacy Survey, hereafter CFHTLS-Wide (Gwyn 2012). This survey covers approximately 170 deg^2 of sky in four patches and uses filters $u*$, g', r', i', and z' with an exposure time of approximately 1 h per filter per field. Atkinson, Abraham & Ferguson (2013, hereafter A13) used visual classifications to study the incidence of faint tidal features in a sample of ∼1800 luminous galaxies drawn from this survey, making it an ideal sample against which to benchmark the performance of CNNs.

The A13 sample contains 1781 galaxies that were selected to lie within the redshift range $0.04 < z < 0.2$ and to have magnitude $15.5 < r' < 17$. These cuts were adopted so as to allow for comparison with previous work on tidal feature classification, to minimize contamination from stars misidentified as galaxies and to limit the sample size to a manageable number for visual inspection. As discussed in A13, this sample is heavily biased towards bright systems, with most galaxies lying in the range $-23 < M_{r'} < -20$ mag. The typical half-light radii of the galaxies is 2–6 arcsec.

The A13 study used thumbnails in the g', r', and i' bands as these were the highest signal-to-noise images. These thumbnails were stacked together to increase contrast. A13 estimate a limiting g-band surface brightness of ≈27.7 mag arcsec^{-2} over small scales. Each stacked image was visually inspected and placed into one of five categories depending on the confidence of the inspector that a tidal feature was present. These ranged from very high confidence of the presence of a feature (level four) to a feature with around 75 per cent certainty (level three) and so on, until very high confidence was reached that no tidal features were present to the depth of the data (level zero). If tidal structure was deemed to be present then it was further classified into six non-exclusive tidal feature classes – shells, streams, miscellaneous diffuse structure, arms, linear features, and broad fans. Roughly 10 per cent of the A13 sample was classified independently by three experts to ensure that the visual classification scheme leads to consistent answers by multiple experts and is therefore reproducible. Following this, the entire sample was classified by a single inspector (Atkinson) in order to maximize consistency. We consider these single expert labels as a ground truth against which to measure automated methods, and address reproducibility in Section 6. The archetypal examples provided by A13 of these feature classes are reproduced in Fig. 1.

As the thumbnails utilized in the A13 study were not available to us, we had to recreate these from scratch, in an identical manner, so as to guarantee that our automated classifier had access to the same information as the human experts. To this end, we extracted

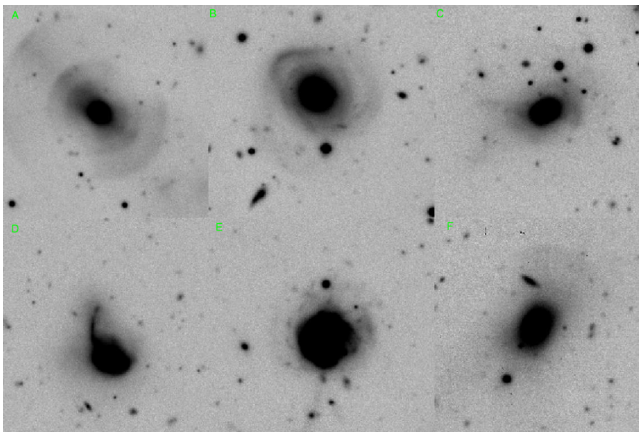


Figure 1. Tidal feature classes defined by Atkinson et al (2013). Clockwise from top left: shell (A), stream (B), miscellaneous diffuse (C), arm (D), linear (E), fan (F). Reproduced by permission of the original authors and the AAS.

256 × 256 pixel regions in the g', r', and i' bands around the galaxy centroid coordinates provided in the A13 catalogue using the CFHTLS cut-out service (Erben et al. 2013). These images were subsequently manipulated in a variety of ways, as will be described in Section 3.2.

To reduce the complexity of the classification problem, tidal confidence labels were binned into binary classes. The choice to restrict the problem to a binary classification was motivated by the limited training data available (see Section 3.4) rather than any fundamental constraint. Non-tidal (0) was matched to confidence ≤25 per cent (levels zero and one in the A13 scheme), whereas tidal (1) was matched to confidence ≥75 per cent (levels three and four in the A13 scheme). Galaxies with a tidal confidence of 50 per cent were deemed to provide no useful information for our purpose and were cut from the sample. Of the 1781 galaxies in the original A13 sample, 24 could not be downloaded in all three bands from the CFHTLS cut-out service, giving an initial data sample of 1757 imaged galaxies. Of those, 1316 galaxies are re-labelled False (non-tidal) and 305 are re-labelled True (tidal). 136 have a confidence of 50 per cent and are therefore removed, leaving a final sample of 1621 galaxies with binary labels. The ability for the method to adapt to more subtle classes given sufficient training data is discussed in Section 6.

3 SINGLE CONVOLUTIONAL NEURAL NETWORK CLASSIFIER

3.1 Introduction to convolutional neural networks

CNNs are a subset of machine learning algorithms frequently used to identify patterns in tensors (i.e. n-dimensional arrays) where the spatial arrangement of values is important. Most commonly, these tensors are the pixel values of images. They routinely show state-of-the-art performance on various image classification benchmarks that require making discerning distinctions between classes and ignoring background effects (Russakovsky et al. 2015). We provide here a brief overview of how these methods work and refer the interested reader to LeCun, Bengio & Hinton (2015), and references therein for detailed descriptions of CNNs, and to Dieleman et al. (2015), Lanusse et al. (2018), and Kim & Brunner (2017) for particular astrophysical applications.

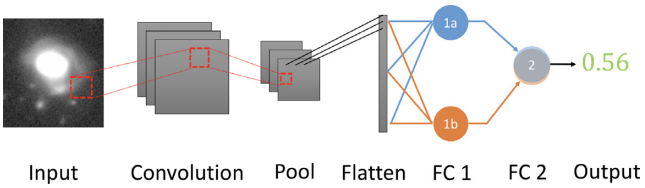


Figure 2. Illustrative diagram of a toy CNN. The pixel values of a galaxy image are taken as input. These are convolved with three filter matrices to create three feature maps. The feature maps are reduced in size by a pooling operation that preserves only the local 2 × 2 maxima, then 'flattened' and concatenated into one dimension. This flattened list of abstract features is the input for two fully connected layers with two and one neurons, respectively. The final fully connected layer outputs a scalar value, to be interpreted as a 56 per cent confidence prediction of the galaxy having tidal features. In practice, the convolutional and pooling operations would repeat several times and the first fully connected layer would include of order 100 + neurons.

Neural networks are composed of repeated tensor operations called layers. The output of layer l, $\mathbf{x}^l$, is the input to layer $l+1$ and the arrangement and connectivity of layers is called the architecture. The net effect of a neural network is a non-linear mapping from input tensor to final layer output (i.e. prediction), with the aim being to learn the mapping which gives the true predictions.

Each type of layer performs a different operation. For example, the most basic is the fully connected layer which performs the operation:

$$\mathbf{x}^{(l)} = f(\mathbf{w}^{(l)}\mathbf{x}^{(l-1)} + \mathbf{b}^{(l)}). \quad (1)$$

Consider the classification of an image using fully connected layers. The image is encoded by the tensor of pixel values $\mathbf{x}^{(0)}$. This input propagates forward through the layers and is modified by the weights $\mathbf{w}^{(l)}$ and biases $\mathbf{b}^{(l)}$ of each layer through repeated operations of equation (1). The output of the final layer is interpreted as predictions for that image.

Networks typically include two additional types of layer: convolutional and pooling. The convolutional layer operation can be described as

$$\mathbf{x}_n^{(l)} = f\left(\sum_i \mathbf{w}_{ij}^{(l)} * \mathbf{x}_i^{(l-1)} + \mathbf{b}_i^{(l)}\right), \quad (2)$$

where $\mathbf{w}_{ij}^{(l)}$ is the filter of layer l. Convolutional layers identify features with a fixed scale relative to the filter size. On the other hand, pooling layers reduce the size of a feature map by aggregation, for example by preserving only the local 2×2 maxima (as in this work). When alternated with convolutional layers, pooling layers allow for features of increasing spatial scale to be detected. Together, convolutional and pooling layers create increasingly abstract feature maps that encapsulate the image content. These features may then be classified using fully connected layers. A toy CNN illustrating each operation is shown in Fig. 2.

The discriminative features to measure from the pixel data, and then to classify, are identified as part of the learning process, described in Section 3.3. This is in contrast to some other machine learning algorithms, such as random forests, which classify using user-defined image features like brightness and asymmetry.

We implement our network using the deep learning library KERAS (Chollet et al. 2015), with TENSORFLOW (Abadi et al. 2015) as a backend.

3.2 Preprocessing

For each galaxy in the sample, the thumbnails in each band were combined and manipulated in a variety of ways before being passed to the classifier. The 'preprocessing' options investigated are listed below in order of operation.

(i) *Aggregation*. The g'-, r'-, and i'-band images provide three tensors of pixel flux values, each of shape (height, width). These are combined to create a single tensor, which includes all pixel information on each galaxy, to be used as input to the network. The bands can be pixel-averaged to create a tensor of shape (height, width). Alternatively, the bands can be concatenated (i.e. placed next to one another) along a third colour dimension to create a tensor of shape (height, width, 3) in analogy with RGB images.

(ii) *Background estimation*. This estimate is required for the pixel intensity clipping and masking procedures described below. To estimate the sky background, we use the functions `sigma_clipped_stats` and `make_source_mask` from the PYTHON package PHOTUTILS (Bradley et al. 2018). `sigma_clipped_stats` estimates background from the statistics of all unmasked pixels within a given σ of the median unmasked pixel value. `sigma_clipped_stats` is called by `make_source_mask` to make an initial background estimate. `make_source_mask` then uses this estimate to detect and mask sources. The masked image is passed back to `sigma_clipped_stats` for an updated background estimate. This procedure iterates five times, giving a final background estimate.

(iii) *Pixel intensity clipping*. The extreme intensity variation between the inner galaxy core and the tidal features can interfere with rescaling algorithms (see below). Retaining only pixels with intensities lower than 6σ above the background avoids this issue.

(iv) *Pixel intensity rescaling*. Rescaling the pixels to reduce the dynamic range of the image ensures that the tidal features contribute to the first layer values. We apply to each tensor x a rescaling mapping, for example $\sinh(x)$, x^a, or $\ln(x)$. Since the values of the first network layer are proportional to the input image pixel values, this avoids the untrained network initially seeing only the bright galaxy cores.

(v) *Masking*. The thumbnails have foreground and background objects, as well as occasional image artefacts, within the field of view. This introduces additional noise that could be mistaken for tidal features by the classifier. To mitigate this, pixels outside the contiguous galaxy light distribution can be masked. To identify which pixels to mask, we use a combination of background estimation and mean convolutions to estimate which pixels are plausibly part of the galaxy. This process is described in detail in Section 5.1.

(vi) *Local smoothing*. This can enhance the appearance of faint tidal features near the signal-to-noise limit, albeit at the cost of a reduction in spatial resolution. We opt to replace each pixel with the local 3×3 average.

As the optimal combination of these various preprocessing options for tidal feature detection is not initially obvious, we approach this problem empirically, by using a grid search – see Section 3.6.

3.3 Training and evaluation

The CNN algorithm consists of two nested loops: an inner training loop and an outer epoch loop. The complete algorithm is illustrated in Fig. 3.

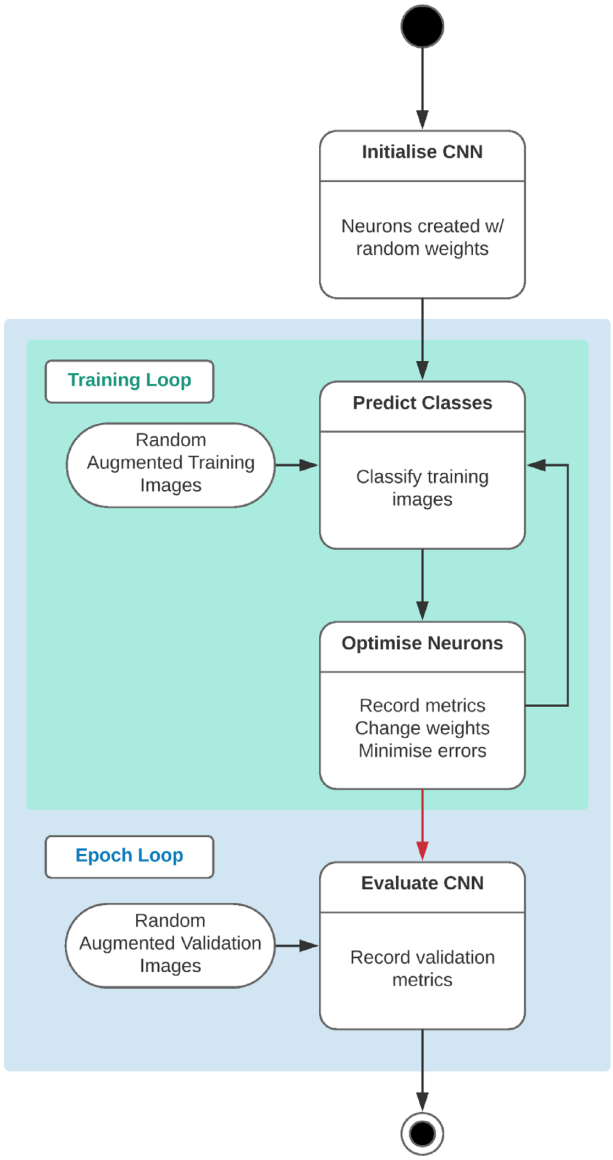


Figure 3. Flow chart of each stage for a single CNN. Red arrows denote steps that only occur after specified iterations have elapsed.

With every iteration of the training loop, the network is gradually fit to the training data. A batch of unique labelled images (see Section 3.5) is given as input to the CNN, and the CNN returns predictions. The quality of these predictions is measured using a loss function. For binary classification problems, a standard choice is the binary cross-entropy

$$\mathcal{L} = -\sum_{i=1}^{N} y_i \log p_i + (1 - y_i) \log(1 - p_i), \tag{3}$$

where the loss is summed over a batch of images of size N, each with a true label y_i and a model score p_i.

The gradient of the loss function with respect to the weights and biases is computed, and the weights and biases are then updated to minimize the loss function. The loop then repeats for a new batch of labelled images. Once a specified number of training loops have elapsed, the epoch loop is executed.

For every iteration of the epoch loop, the network makes predictions for a batch of 'unseen' validation images not used in the

training process. Metrics for the quality of these predictions (for example, the mean accuracy) are recorded. The training process (i.e. multiple training loops) is then restarted with a new configuration. Once a specified number of epoch loops elapse, the algorithm terminates.

All figures in this work use a batch size of 75 images, identified as the optimal number by the grid search (see Section 3.6). One epoch is arbitrarily set as 14 batches or 1050 training images. Batch images are randomly selected without replacement (i.e. selected only once) in equal proportion from the tidal and non-tidal galaxy training subsets. Once all images from a subset have been selected once and removed, the subset is refilled. First, this approach provides the network with sufficient tidal examples to learn from. Secondly, it allows the network prediction to be interpreted as the probability that a given image is tidal and not merely a reflection of the base rate (i.e. the relative number of tidal versus non-tidal galaxy training examples seen by the network). Excluding the base rate during training ensures that predictions on a new sample will not be biased towards the training base rate. Each selected image is randomly transformed to augment the data set (see Section 3.5).

Any initial partitioning of data into training and validation images is arbitrary; we could have selected any set of images as validation images. We therefore need to check if the network is fortuitously performing better on those validation images than it would on a large set of new data. Smaller data sets are particularly susceptible to such accidental overperformance, as small number statistics make this scenario more likely. We use five-fold cross validation to ensure our prediction quality metrics do not depend on an arbitrary division of data into training and validation subsets. In n-fold cross-validation, the complete data sample is split into n random subsets. $n-1$ are used to train the classifier from scratch, and the remaining subset is used as validation data. This is repeated for all n permutations. All prediction quality metrics in this work are then averaged from each of the five-fold cross-validation runs.

3.4 Network architecture

A significant challenge with our CNN approach to tidal feature identification is our exceptionally small training sample. Because every neuron connection is assigned a weight, CNNs typically have $>10^5$ free parameters to learn (i.e. to fit to the data). Having many free parameters allows the learning of more complex features, but increases the entropic capacity of the classifier. Without a correspondingly large training sample to provide constraints, overfitting occurs and degrades the performance. CNNs are typically applied to samples of 10^4–10^{10} images – see e.g. Simonyan & Zisserman (2014), Dieleman et al. (2015), Huertas-Company et al. (2015), Kim & Brunner (2017), and Petrillo et al. (2017) – while our CFHTLS-Wide sample contains only 1621 galaxies, of which a mere 305 have tidal features. We therefore need to operate approximately two orders of magnitude below the minimum sample sizes normally used by CNNs.

We initially choose the architecture by Chollet (2016), a relatively modest network architecture with (only) 3714 593 free parameters (see Fig. 4) designed for smaller data sets. Convolutional layers have Fn (e.g. F32) 3×3 convolutional matrices (i.e. filters), each with a convolution step size (i.e. stride) of 1×1. Fully connected layers have Nn (e.g. N64) neurons. The final layer is a single neuron whose output represents the continuous-valued class prediction.

We verify with a grid search that this architecture outperforms three convolutional layer networks with significantly higher or lower numbers of convolutional filters or layers. Three convolutional layers provide enough depth for high performance without becoming computationally intractable, while the relatively low number of filters helps prevent overfitting. The majority of free parameters (3686 464) are in the first fully connected layer.

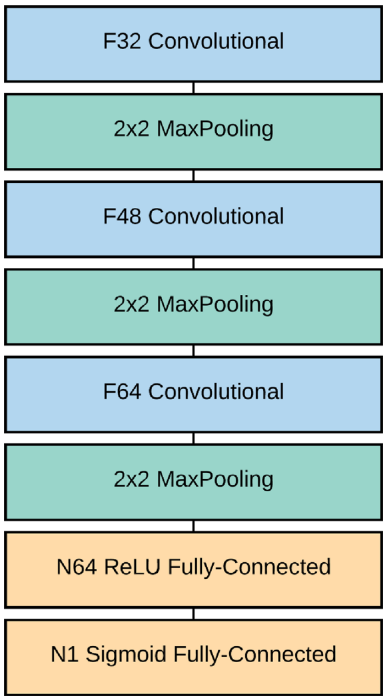


Figure 4. Network architecture for single CNN. Input image (tensor) at top. Output prediction at bottom.

To further minimize overfitting, we apply 'dropout' to this layer. Intuitively, dropout temporarily removes random selections of neurons. This encourages neurons to learn parameters that remain discriminative for many different combinations of other neurons in the network. A neuron and all associated connections are referred to as a unit. For each training epoch, each unit in the fully connected network layer has probability p to be removed for that epoch. The operation of a fully connected layer with dropout is

$$x_i^{(l)} = f(\mathbf{w}_i^{(l)}\mathbf{x}'^{(l-1)} + \mathbf{b}_i^{(l)}), \tag{4}$$

where $\mathbf{w}_i^l$ denotes unit i in layer l, $x_i^{(l)}$ is the (scalar) output of unit i in layer l, and $\mathbf{x}'$ is the elementwise product $\mathbf{x}' = \mathbf{x}\ (*)\ \mathbf{B}(p)$ with $\mathbf{B}(p)$ being an $\mathbf{x}$-shaped matrix with binary elements according to the Bernoulli distribution (i.e. 1 with probability p, 0 with probability $1-p$).

The thinned network (following dropout) is trained for a single epoch before $\mathbf{B}(p)$ is re-evaluated, causing different units to be active and a new thinned network to be created (sharing weights with the predecessor). Dropout therefore effectively trains many unique networks, increasing prediction quality. We select the hyperparameter p to be 0.5, based on the results of the grid search described below.

3.5 Augmentation

Galaxy morphological classifications should be invariant under certain transformations, such as flips, rotations, minor zooms, and minor translations. CNNs lack our intuitive understanding of transform invariance and require sufficiently diverse examples to infer which transforms are not discriminative. We therefore artificially

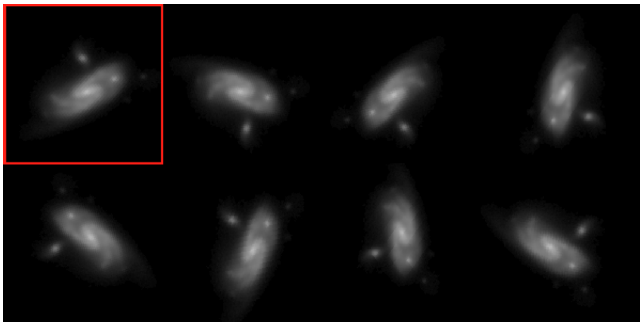


Figure 5. Mosaic of illustrative augmented images of a single non-tidal galaxy. Images are mean-averaged, masked (3σ), and shown in log scale. The images are cropped from 256 to 150 pixels for illustration only. The original image is shown in the top left (red highlight).

expand our training set by including many variants of the original input images. By inputting many randomly transformed images with unchanged labels, we teach the network to be insensitive to those transforms. By applying these transforms dynamically when each input image is read by the network, the effective training set becomes arbitrarily large and the network always sees a unique image. Note that augmented images are less informative than truly new images; once the network has learned the invariance, further augmented images do not improve performance.

We randomly apply each of the following transforms to augment the images:

(i) Horizontal flip.
(ii) Vertical flip.
(iii) $\pm\frac{\pi}{2}$ resampled rotation.
(iv) 90 per cent to 110 per cent resampled zoom.
(v) $\pm$5 per cent horizontal translation.
(vi) $\pm$5 per cent vertical translation.

To avoid unnecessary information loss from pixel resampling after each step, the transforms are applied through a single net transformation. We verify with a grid search that the resolution degradation from the net transformation has a less significant impact on prediction quality than the corresponding learned invariance.

Fig. 5 shows a single galaxy with seven different augmentations applied. The same random augmentation process creates a unique image each time.

3.6 Grid search

CNNs have tuneable design values (e.g. layercount = 4, firstlayer-width = 256) called hyperparameters (Lu et al. 2015). The choice of hyperparameters may have a significant impact on classifier performance, but the optimal choice is not known a priori. Estimates can be made with heuristics (rule-of-thumb guesses) based on previous generic image classification work. However, images of galaxies with faint tidal features are unlike 'terrestrial' pictures in that they have extreme contrast, high noise levels and indistinct subject shapes, and so borrowing from such work is unlikely to be optimal.

We improve our hyperparameter estimates using grid searches. Through this procedure, many possible network configurations are trained and measured. We choose to separate hyperparameters into related groups and then identify the optimal choice within that group through an exhaustive grid search. For example, we assume the optimal number of layers is independent of pixel rescaling and proceed to test many possible numbers of layers with a single rescaling. This approach makes the grid search computationally feasible without needing to specify any hyperparameters with heuristics.

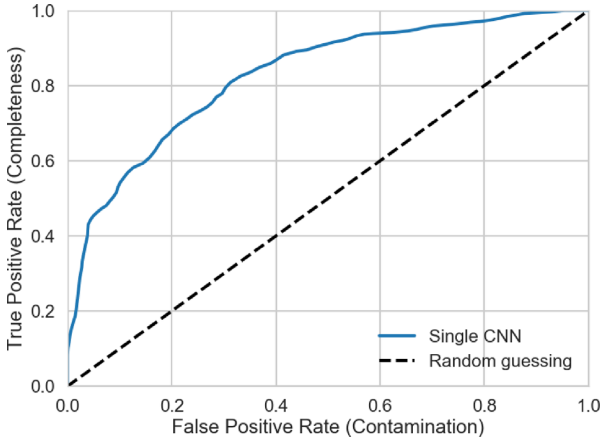


Figure 6. The ROC curve for a single CNN classifier on CFHTLS-Wide images. The dashed line indicates the expectation for random guesses.

We use three groups of hyperparameters: preprocessing (see Section 3.2), architecture (see Section 3.4), and augmentation (see Section 3.5). We find that the best performing preprocessing configuration is band-stacked images with 3σ masking, logarithmic pixel intensity scaling, and no mean convolutions. Performance is invariant under physically reasonable choices of pixel clipping values and, provided that mean convolutions are not used, also invariant under pixel intensity rescaling. This latter result is intuitively surprising given the impact that rescaling has on human perception. The best performing architecture and augmentation configurations have already been discussed in Sections 3.4 and 3.5, respectively.

3.7 Single network results

We select completeness and contamination as metrics to evaluate the performance of our network. Conceptually, completeness is the probability for a visually classified tidal galaxy to be correctly identified by the CNN as tidal, and contamination is the probability that a visually classified non-tidal galaxy is incorrectly identified by the CNN as tidal. Mathematically, these are the true positive rate (TPR) and false positive rate (FPR), respectively,

$$TPR = \frac{TP}{TP + FN} \qquad FPR = \frac{FP}{FP + TN}. \tag{5}$$

The Receiver Operating Characteristic (ROC) curve illustrates the completeness and contamination of the classifications. The ROC curve of our best-performing single CNN classifier is plotted in Fig. 6. The completeness and contamination may be selected along any point on the curve, corresponding to varying the confidence threshold used to classify images as tidal. For example, one might choose a completeness of 70 per cent and therefore a contamination of 22 per cent. Random guessing would provide equal completeness and contamination.

Fig. 7 shows the accuracy of a single classifier with and without dropout and augmentations, averaged over five runs. Shaded regions denote the 90 per cent Bayesian credible interval (Oliphant 2006). Without dropout and augmentations, the training accuracy increases with the number of galaxies that the network sees while the validation accuracy remains low. This is because the network is overfitting to random patterns in the training data. These patterns do not generalize to new data so the validation accuracy remains

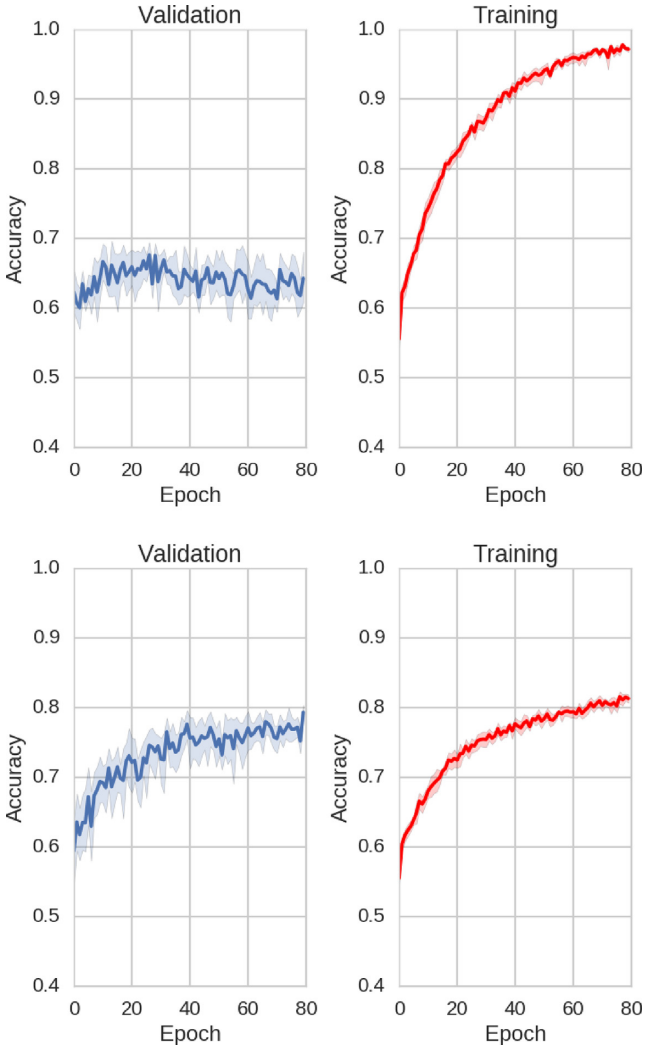


Figure 7. Mean training metrics for the same network architecture with dropout and augmentations off (top) versus on (bottom), over five runs (trained and validated on each five-fold cross-validation permutation). Shaded regions denote the 90 per cent Bayesian credible interval.

low. In contrast, with dropout and augmentations, the network is learning patterns present in both the training and validation data, causing both accuracies to rise.

Fig. 8 shows performance broken down by class of tidal feature, following the schema introduced by A13. Every prediction is made by a network which has not been trained on that galaxy, following the cross-validation strategy described in Section 3.3. Networks perform best (i.e. have the lowest mean absolute error) on fan features, a surprising result given the relative rarity of such features. In general, performance is higher for dispersed features (fan, diffuse, shell) than small-scale structural features (arm, stream, linear). We speculate that this may be because such features are unlikely to be mimicked by contaminant objects in the field of view, and therefore easier to learn from our relatively small data set.

All classes except fan (which is both rare and has a low mean error) have at least one prediction with an error close to one. This reflects the probabilistic nature of the method; statistical metrics of success do not imply that every prediction is approximately correct. Fig. 9 shows the galaxies with the highest and lowest absolute error (matching the highest and lowest horizontal bars across all

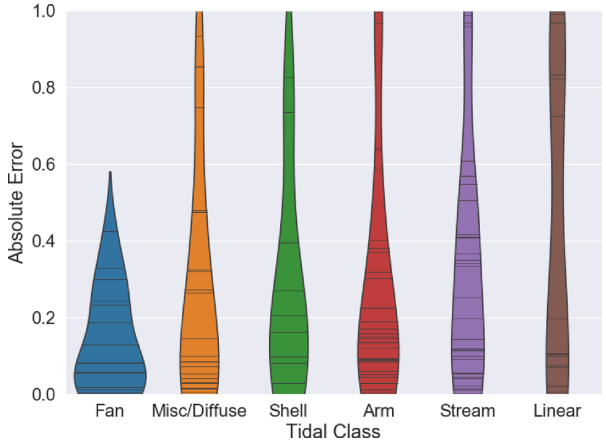


Figure 8. Single network validation performance by class of tidal feature. Each column is a class of tidal feature, ordered left to right by increasing mean absolute error on galaxies with that feature. Horizontal black bars denote individual galaxies: for example, a galaxy with a fan feature on which the network prediction had an absolute error of 0.2. More galaxies with lower absolute error indicates better performance. The area of each column illustrates the probability density, inferred (by kernel density estimate) from all galaxies of that class. Feature classes follow the schema introduced by A13.

columns in Fig. 8). Failures show no obvious pattern, underscoring how the operation of convolutional neural networks is not always immediately interpretable by humans. We investigate the behaviour of the network in Section 6.1.

4 ENSEMBLE CLASSIFIER

4.1 Configurations

The predictions of an ensemble of classifiers will typically outperform those of a single classifier because each independent prediction provides new information on the classification of the input image. This information is exploited through averaging over the individual predictions (e.g. Zhang & Ma 2012). Indeed, ensemble methods are routinely used to improve image classification performance, e.g. Dieleman et al. (2015). We investigate two different ensemble configurations – CNN using optimal preprocessing (configuration A), and CNN using varied preprocessing (configuration B) – as a means to generate more accurate faint tidal feature classifications for our sample.

In configuration A, each CNN is in the optimal hyperparameter configuration identified in Section 3. The random order of input training images and the random initialization of weights and bias prior to training may cause the CNN to converge to different local minima during training, particularly when applied to smaller training sets (LeCun et al. 2015). This leads to identically configured CNNs making slightly different predictions, which is described as stochastic independence.

In configuration B, each CNN uses varied preprocessing hyperparameters, as detailed below. This introduces further independence between CNNs. Different preprocessing hyperparameters might lead a CNN to advantageously detect different tidal features. For example, more restrictive masking thresholds will reduce the number of contaminant objects in the field of view but may also reduce the spatial extent of particularly faint tidal features. However, hyperparameters that are different to the optimal hyperparameters will degrade the performance of a single model. By comparing each

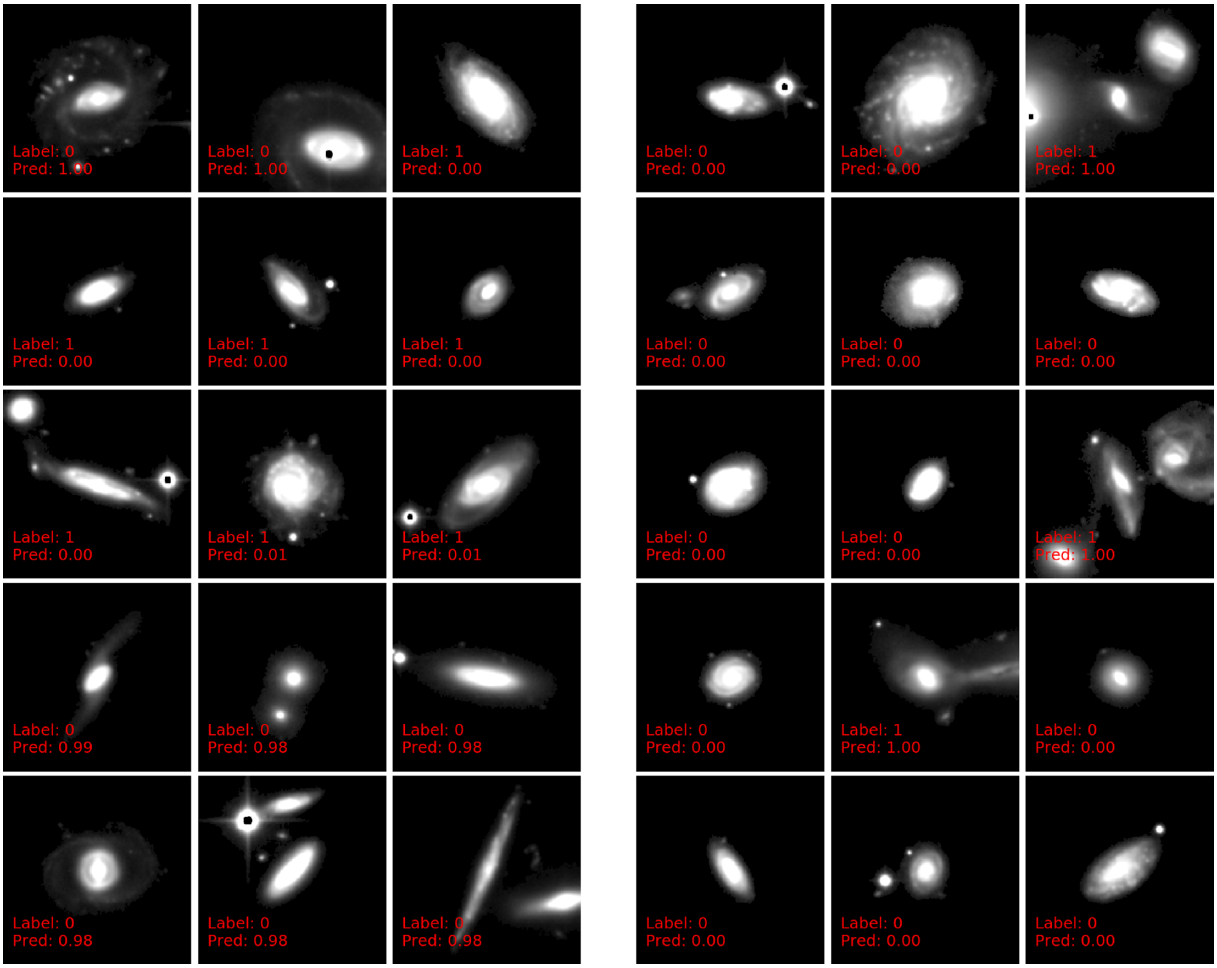


Figure 9. Galaxies with the highest (left) and lowest (right) absolute error in validation predictions, as presented to the network following the preprocessing strategy identified as optimal (including pixel rescaling and background masking, see Section 3.2). Brightness and contrast have been further adjusted for human viewing of tidal features.

ensemble configuration, we test if (for our problem) it is more effective to ensemble individually stronger classifiers with lower independence (configuration A) or individually weaker classifiers with higher independence (configuration B).

We select the following set of preprocessing hyperparameters for the five CNNs comprising the configuration B:

(i) Logarithmic rescaling, 3σ mask threshold (i.e. optimal).
(ii) Logarithmic rescaling, 5σ mask threshold.
(iii) No (linear) rescaling, 3σ mask threshold.
(iv) No (linear) rescaling, 5σ mask threshold.
(v) No (linear) rescaling, band-stacked (un-masked) image.

These were chosen for being high-performing combinations identified with the grid search described in Section 3.6, and for spanning visually distinct preprocessing steps.

4.2 Training and evaluation

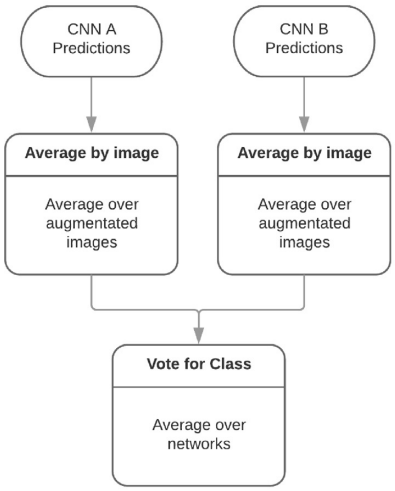
To decide which configuration has the best performance, we need to train and evaluate all five CNNs composing that configuration. Each CNN is trained on images that are randomly drawn in equal measure from 80 per cent of the tidal and non-tidal classes, as described in Section 3.3. Each CNN is then asked to make several predictions on each galaxy in the remaining 'unseen' 20 per cent. We then calculate an overall prediction for the configuration by combining the predictions of each CNN. Fig. 10 illustrates how the predictions of each CNN are combined.

First, for each CNN, we average over all predictions made by that CNN on augmented images of the same galaxy. We know that the true label is invariant under our augmentations but the CNN may not have completely learned to ignore them. Averaging over predictions of the same galaxy ensures that our final configuration prediction will not depend on any particular augmentation.

After recording the augmentation-averaged prediction on each galaxy by all five independently trained CNNs, we then average those single-CNN predictions that allow us to exploit any independence in those predictions to improve performance, as explained in Section 4.1.

4.3 Results

Fig. 11 shows the average ROC for each ensemble configuration, and overplots the ROC of the individual optimal CNN shown in Fig. 6. We find that both ensemble configurations provide a significant improvement over using a single optimal CNN, as expected. For example, with a single CNN we were able to achieve a completeness of 70 per cent with a contamination of 22 per cent. This



Galaxy	CNN A Prediction	CNN B Prediction	True Label
a	0.86	0.71	1
a	0.83	0.82	1
a	0.92	0.81	1
b	0.43	0.35	0
b	0.54	0.39	0
b	0.32	0.42	0

Galaxy	CNN A Prediction	CNN B Prediction	True Label
a	0.87	0.77	1
b	0.46	0.39	0

Galaxy	Prediction	True Label
a	0.86	1
b	0.43	0

Figure 10. Flow chart of each stage for ensemble classifier. Red text illustrates the values being combined at each stage.

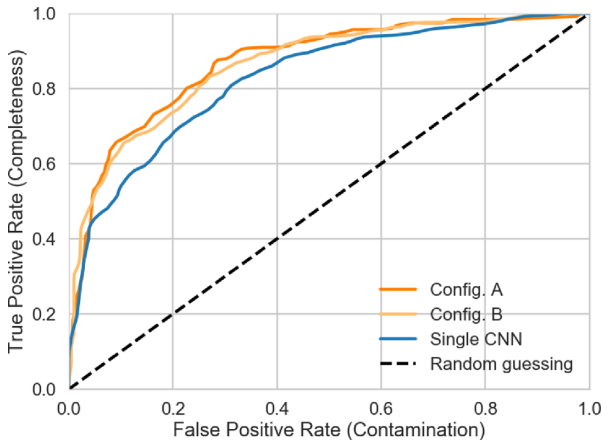


Figure 11. Comparison of the ROC curves of a single CNN, and our two ensemble CNNs.

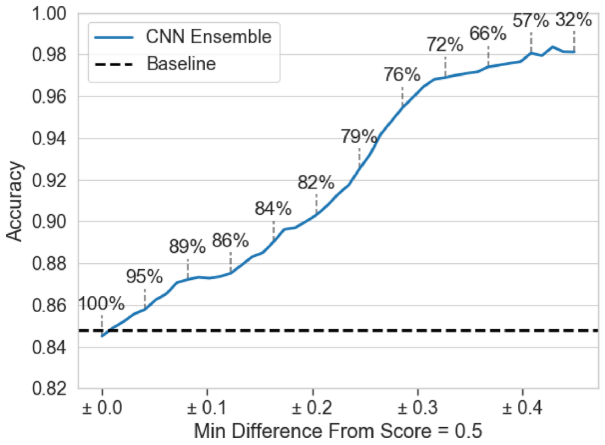


Figure 12. Accuracy of CNN ensemble (configuration A) on subsamples with increasingly confident predictions. Accuracy is measured for galaxies where the classifier score is a given minimum difference from 0.5. The greater the minimum difference, the more confident the classifier is. The percentage of galaxies with at least that confidence is annotated. For example, 72 per cent of galaxies have a minimum score difference of at least ±0.33 (i.e. a score above 0.83 or below 0.17) and can be classified with 97 per cent accuracy. Also shown is a baseline classifier that always predicts non-tidal (the majority class).

now improves to a completeness of 76 per cent ± 2 per cent with the same level of contamination. The prediction quality of the two ensemble configurations is approximately equal within the expected statistical variation. That is, for our problem, both configurations are equally effective.

The ROC curve measures performance when classifying all galaxies, which is appropriate for understanding the overall performance of a method. In practice, we might instead choose to classify only a subset of galaxies for which the model is reasonably confident, and refer the remainder to experts or citizen scientists. We can measure model confidence using the continuous prediction score output by the model. By optimizing our model using the binary cross-entropy loss (equation 3), which heavily penalizes mistaken scores near 0 or 1, we can interpret scores near 0 or 1 as confident predictions and scores close to 0.5 as uncertain predictions (Tewari & Bartlett 2005). Therefore, we can select galaxies with confident predictions by requiring a score at least some minimum difference from 0.5.

However, because the model was trained on an equal number of tidal and non-tidal galaxies (Section 3.3), our scores on the full imbalanced sample are uncalibrated; the model does not know that non-tidal galaxies are common. To account for this, we calibrate our scores with Platt's Scaling (Fonseca & Lopes 2017). We use logistic regression to fit a correction to the fraction of true positives on 25 per cent of galaxies, such that the scores match the empirical probability that a galaxy is tidal, and then apply that correction to the scores of the remaining galaxies.

Having calibrated our scores, we can now measure how performance varies on increasingly confident subsamples. We find that performance can be dramatically improved, at the cost of leaving some galaxies unlabelled. Fig. 12 shows how the accuracy varies as we classify only galaxies where the model is increasingly confident. On the full sample, the calibration causes the model to predict 'non-tidal' on three out of four galaxies, leading to an accuracy similar to a baseline classifier that always predicts non-tidal. How-

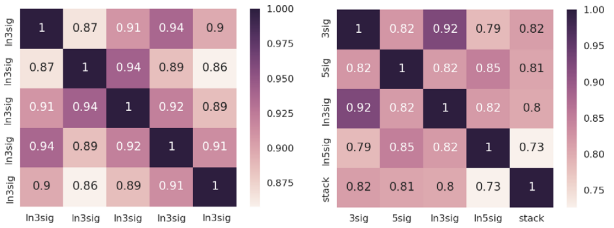


Figure 13. Pearson correlation coefficients between the predictions of single CNN (rows, columns) classifiers acting within ensembles using optimal preprocessing (A, left) and varied preprocessing (B, right). Labels denote the preprocessing used for that CNN, with 'ln' denoting logarithmic pixel rescaling and 'Nsig' denoting the masking threshold used. Configuration A combines five classifiers all using the same optimal preprocessing configuration (logarithmic rescaling and a 3σ pixel mask) while configuration B combines five classifiers with varied preprocessing configurations.

ever, by using the score to identify galaxies where the model is more confident, we can make useful predictions on the bulk of the sample. For example, the 72 per cent of galaxies with a minimum score difference of at least ± 0.33 can be classified with 97 per cent accuracy, compared to 84 per cent accuracy on all galaxies. This suggests that our prototype model can be used to identify the bulk of a survey with near-perfect accuracy, drastically reducing the human labelling effort required to create extensive science-ready catalogues of galaxies with or without tidal features.

We next investigate the independence of the single classifiers within each ensemble by measuring the correlation between each possible pair of classifiers.

The correlation is measured with the Pearson r correlation coefficients between the continuous-valued predictions of each classifier. The resulting matrices are shown in Fig. 13 and are symmetric due to the symmetry of the correlation coefficient. Unitary diagonal elements result from pairwise comparisons between a CNN and itself, and may therefore be neglected.

Recall that configuration A combines five classifiers all using the same optimal preprocessing configuration (logarithmic rescaling and a 3σ pixel mask, see Section 3.6), while configuration B combines five classifiers with varied preprocessing configurations. The average (non-unitary) correlation coefficient is lower for the ensemble with varied preprocessing (B, $\bar{r} = 0.82$) than with optimal preprocessing (A, $\bar{r} = 0.90$), indicating that *additional independence can be introduced by altering the preprocessing process*. In particular, altering the masking threshold has a greater effect on classifier predictions than changing from logarithmic to linear rescaling. This is consistent with our earlier finding that prediction accuracy is invariant within statistical uncertainty under pixel rescaling.

5 COMPARISON WITH CURRENT METHODS

As discussed in Section 1, most current methods of automated feature detection are not well suited to recovering the typical low surface brightness tidal features that arise from minor mergers and accretions. We have selected two promising alternative methods from the recent literature and applied them to the A13 sample in order to benchmark their performance against that of the CNNs. These are as follows:

(i) Shape asymmetry (Pawlik et al. 2016), an example of a method based on non-parametric feature extraction;

(ii) WND-CHARM (Shamir 2012; Schutter & Shamir 2015), an alternative unsupervised machine learning approach previously shown to be successful in identifying peculiar and interacting galaxies.

Detecting tidal features by any method is dependent on the following:

(i) The nature of the sample under study. The varying depths, bandpasses, and spatial resolutions of different data sets can lead to incomparable detection rates;

(ii) The author's definition of what is tidal. The context of the paper often sets the definition for a tidal feature, and different authors may reasonably have different definitions.

For example, Bridge, Carlberg & Sullivan (2010) and A13 both use data from the CFHTLS to identify tidal features through visual inspection. However, Bridge et al. (2010) use data from the Deep component of the survey, which covers less sky area but is sensitive to more distant galaxies than the Wide component used by A13. Furthermore, they select different features to define which galaxies are tidal (tidal tails and bridges versus the more subtle debris features outlined by A13). Directly comparing the detection rates (and underlying methodology) of these two papers is therefore not meaningful as they measure different things.

Through applying all three methods to the same galaxy sample, with the same binary labels, we sidestep many of the complications that arise when comparing results that have appeared in the literature. We also ensure that the ability of each classifier to detect *the same* tidal features is tested fairly. Below, we describe each method and motivate why we have selected that particular method for comparison.

5.1 Application of the shape asymmetry method

Shape asymmetry was introduced by Pawlik et al. (2016) as a method to automatically detect faint asymmetric tidal features in galaxies that experienced a recent merger. It is an appropriate choice for tidal feature detection in galaxies with complex morphologies since, unlike residual-based methods, it does not require a parametric fit of the underlying galaxy light profile. The measure is only sensitive to morphological asymmetry and does not contain information about the asymmetry of the light distribution. When applied to a sample of 70 starburst and post-starburst galaxies imaged by the Sloan Digital Sky Survey (Abazajian et al. 2009), Pawlik et al. (2016) report an accuracy of 95 per cent in detecting post-merger tidal features.

The method works as follows. First, following Conselice (2003) and Conselice (2014), the minimal asymmetry centroid is identified and asymmetry parameter A is recorded.

$$A = \frac{\sum |\, I_0 - I_{180}\,|}{2\sum |\, I_{180}\,|} - A_{\rm bgr}, \tag{6}$$

where I_0 is the value of a galaxy pixel, I_{180} is the value of the pixel at the same position after the image is rotated 180 deg, $A_{\rm bgr}$ is the estimated contribution to asymmetry from background noise, and all sums act over all pixels. Note that low surface brightness pixels will have small I_0 and hence will make only minimal contributions to A. As a result, A is relatively insensitive to faint tidal features.

Next, a 3×3 mean convolution is applied to the galaxy image to enhance low surface brightness features. A binary mask is then created with values of 1 where the corresponding pixel count is both some chosen $N\sigma$ above the original measured sky background and

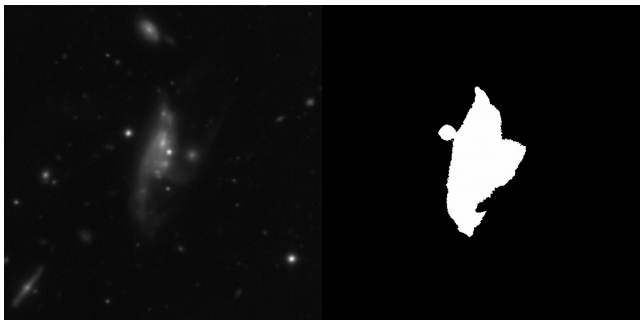


Figure 14. Illustration of the non-parametric shape asymmetry method of Pawlik et al. (2016), applied to a galaxy in the CFHTLS-Wide sample. Left: stacked *gri* galaxy image (logarithmically rescaled for illustration only). Right: binary mask of pixels above 3σ used to calculate shape asymmetry A_s.

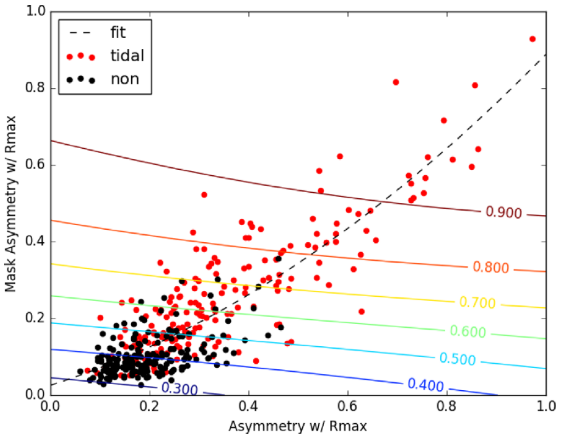


Figure 15. Probability space generated by Pawlik method on 500 CFHTLS-Wide galaxies from the A13 sample, illustrated by contours. Mask asymmetry is the shape asymmetry A_s. Galaxies are observed to follow a clear linear trend on the mask asymmetry/asymmetry space, which we fit for interest only.

contiguously eight-connected to the central pixel, and 0 elsewhere. The intuitive effect is to create a silhouette of the galaxy outline that includes faint structure – see Fig 14. For our implementation, background estimation is done with the procedure described in Section 3.2. We find a pixel masking threshold of $N = 3$ gives optimal results.

Finally, the shape asymmetry parameter A_s is calculated in analogy to A but with I_0 and I_{180}, replaced by the pixel values of the binary mask, rather than the original galaxy image:

$$A_s = \frac{\sum | M_0 - M_{180} |}{2 \sum | M_{180} |}, \tag{7}$$

where M (M_{180}) is the value of a mask pixel at some (rotated 180 deg) position on the binary mask.

To ensure tidal features at the image extremities are included, the selection radius used to calculate both A and A_s is taken as the minimum radius that encloses the full binary mask. By plotting A against A_s, an empirical selection cut can be made to identify galaxies with tidal features.

Fig. 15 shows the resulting asymmetry space for our CFHTLS-Wide sample where 250 random examples are plotted per binary class. On the basis of visual inspection, Pawlik et al. (2016) chose an empirical cut of $A_s > 0.2$ to select tidal galaxies. However, we would prefer to understand how the shape asymmetry method balances completeness and contamination. To do this, we generate ROC curves using two methods (each generating a slightly different curve). In the first method, we generalize the $A_s > 0.2$ sample cut by making many sample cuts separated by δA_s. The ROC curve is then calculated in the continuous limit $\delta A \to 0$. In the second method, we divide the galaxies into five subsets, train a logistic regression classifier (Pedregosa et al. 2012) implemented in `scikit-learn` on four subsets, and make predictions on the remaining test partition. This is repeated for each combination of partitions (i.e. five-fold cross-validation). The ROC curve is calculated as the mean ROC curve over the test predictions for combination. To verify that the logistic regression classifiers are functioning correctly, we illustrate the mean estimated tidal probabilities at every point with contours on Fig. 15.

5.2 Application of the WND-CHARM algorithm

Here, we explore the general-purpose image classification algorithm WND-CHARM (Orlov et al. 2008). It provides an example of unsupervised machine learning and is publicly available as both a command-line tool and PYTHON API from https://github.com/wnd-charm/wnd-charm.

Like CNNs, WND-CHARM was originally developed for other uses (Orlov et al. 2008) and was only later applied to astronomy. It has been successfully used to classify peculiar galaxies (Shamir 2012) and general galaxy morphology (Schutter & Shamir 2015) and so could be reasonably expected to perform well on the problem of faint tidal debris. WND-CHARM represents a middle ground between algorithms that use user-defined image features (for example, random forests – see Ball & Brunner 2009) and algorithms that infer the ideal features to construct from pixel data (for example, CNNs).

WND-CHARM (Orlov et al. 2008) identifies the macroscopic properties (e.g. total image brightness) that are most discriminative between classes in the training sample. Those properties are then used to classify test images by identifying the class of the closest known example, with 'closest' defined in a multidimensional Euclidean space where each dimension corresponds to a discriminative property.

The augmentation procedure we use for our convolutional network (see Section 3.5) is designed to improve classifier performance. To provide a fair comparison, we train and test WND-CHARM on subsets of 25 000 images preprocessed and augmented through the same procedure. We use a train-test split of 80 per cent and 20 per cent, respectively, to evaluate the algorithm.

5.3 Overall comparison

Fig. 16 shows the completeness and contamination achieved by the three approaches over many confidence thresholds (not shown). Fig. 17 summarizes overall performance with the area-under-curve (AUC) scores for each method. The AUC score is frequently used in machine learning literature as a scalar summary metric of classifier quality (Huang & Ling 2005). For our problem, the AUC score measures the probability that a random galaxy with tidal features will correctly be recognized as being more likely to have such features than another random galaxy without tidal features.

It is readily apparent that our CNNs have higher completeness and lower contamination than either of the alternative methods investigated in this paper. The ensemble configurations show the best

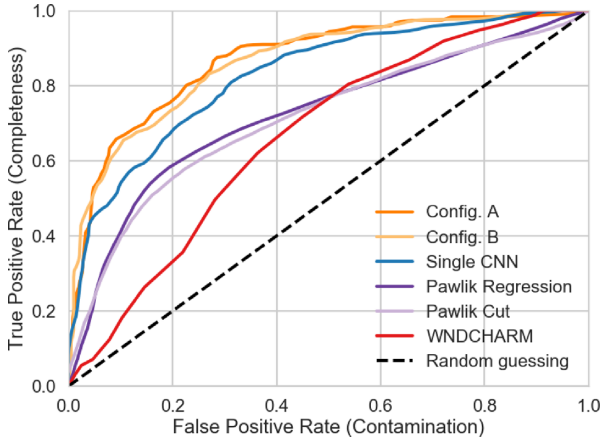


Figure 16. The ROC curves for all classifiers tested on the A13 sample.

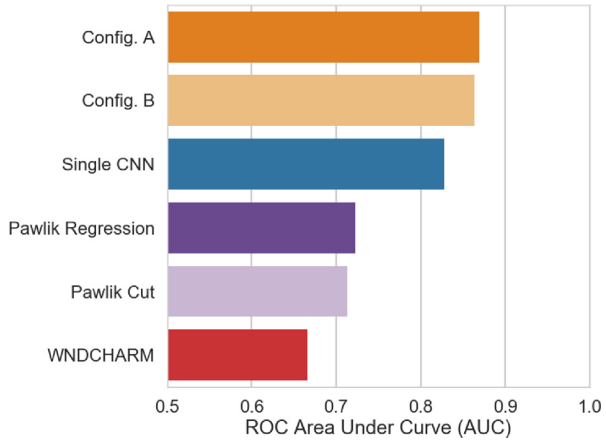


Figure 17. The ROC area-under-curve (AUC) values for all classifiers tested on the A13 sample.

overall performance, followed by the single-classifier configuration. Of the alternative methods, shape asymmetry outperforms WND-CHARM. All the methods tested definitively outperform random guessing.

The completeness and contamination of the two CNN ensemble configurations are notably improved over the single CNN for the more challenging half of tidal galaxies, leveraging residual independence between classifiers to increase performance. For the clearest half of tidal galaxies, the network ensembles show relatively little improvement. This could be a consequence of there being relatively little disagreement between ensemble classifiers for the most obvious examples, reducing the impact of interclassifier voting.

We deem the shape asymmetry method to be moderately effective in identifying galaxies with faint tidal structure. For example, Fig. 16 shows that it can achieve a completeness of 58 per cent with a contamination of 20 per cent. As shown in Fig. 15, there is generally a decent separation between tidal and non-tidal galaxies in A_s versus A parameter space. Galaxies with high values of both A_s and A are likely to be tidal, and for extreme values of these parameters the tidal prediction can be made confidently. This leads to the sharp gradient (corresponding to a rapid rise in completeness while preserving a low contamination) observed in Fig. 16 for contamination <0.2. However, the shape asymmetry method performs less well for galaxies with moderate A_s and A, causing the gradient to subsequently flatten as less confident predictions are included. Extending shape asymmetry to use logistic regression rather than cuts provides only a small improvement.

We also find that WND-CHARM is the least effective method investigated for identifying galaxies with faint tidal features in CFHTLS-Wide sample. We speculate that its poor performance may be a consequence of the macroscopic feature extraction step employed in the algorithm. The ratio of image information content corresponding to faint tidal features may be sufficiently low that WND-CHARM struggles to identify a genuinely predictive feature set amongst the 'noise' of the general morphology. With an ability to investigate up to 4027 image features for correlations with labels, WND-CHARM could be overfitting to image features that do not relate to tidal features in the test data. However, WND-CHARM does show impressive performance on low confidence predictions, even exceeding that of shape asymmetry for contamination $\lesssim 0.5$. Ironically, this suggests that WND-CHARM is able to detect indications of tidal features on the more challenging images while it struggles to confidently detect such features on the clearest examples.

6 DISCUSSION

6.1 Heatmaps

A common criticism of CNNs, and deep learning in general, is that they are 'black box' algorithms which are difficult to interpret. While the resultant classification is readily apparent, the way in which it was arrived at is usually less so. There is no clear link from the properties of the galaxy features to the prediction made.

In order to establish if our method is really identifying faint tidal features in the way we intend, we use prediction heatmaps (Zeiler & Fergus 2014). Having established that each ensemble offers comparable performance, we arbitrarily investigate Configuration A (similar individual classifiers).

For a single image, we inject a synthetic low surface brightness tidal structure into a small area. First, we create a 5×5 grid of pixel values from a Gaussian distribution with background variance and a mean 3σ above the background which represents the synthetic structure. Secondly, we take the original image and replace a random 5×5 pixel area with our new structure.

Each time we add the structure, we reclassify the new image (original plus synthetic structure) with an ensemble classifier and record the change in tidal prediction from the original image. By plotting the tidal predictions as a heatmap where each pixel is the tidal prediction given a 5×5 synthetic structure at the location, we can identify in which image regions the ensemble sensitive to small changes. We assume that adding a tiny synthetic structure to a region that the network prediction is highly sensitive (one might say, 'suspects' as being tidal) causes a much greater increase in the tidal prediction for the whole image than adding such structure to an otherwise non-tidal region.

Fig. 18 shows one example. The input image is shown at the top left. After a brief (5 epoch) training period, the heatmap is approximately a pixel-count-weighted distribution. After training is complete (epoch 125), the heatmap shows the network to have identified a linear feature at the bottom left corner of the image. Redisplaying the original image on a logarithmic scale, we verify that there is indeed a low surface brightness linear feature present at that location. This feature is detected and localized by the network

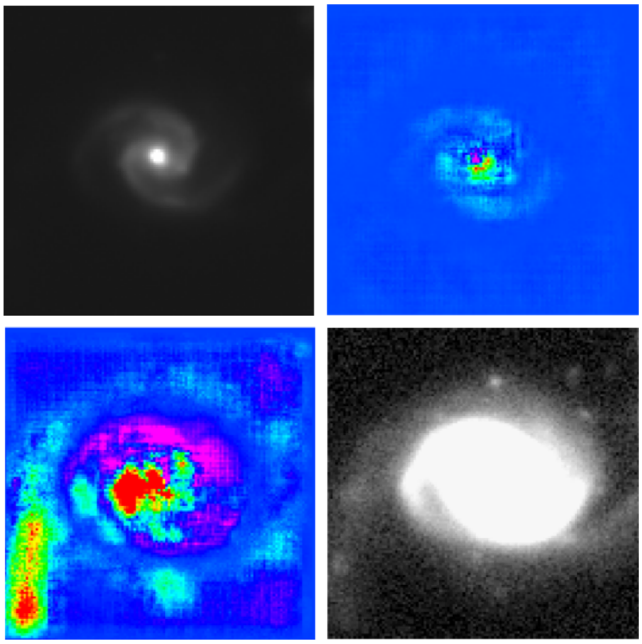


Figure 18. Top left: a cleaned galaxy image without rescaling. Top right: the heatmap from epoch 5. Bottom left: the heatmap from epoch 125. Bottom right: cleaned image with logarithmic rescaling. Magenta denotes non-tidal. Blue denotes neutral. Green through yellow through red denotes increasing tidal confidence. Note that the synthetic tidal structure is only added temporarily to alter the network predictions, and is not shown in any of the images above.

despite being sufficiently faint to be invisible to the eye on the unscaled input image.

Our prediction heatmap demonstrates that the CNNs are identifying which image pixels are associated with low surface brightness tidal features. If the pixel associations are sufficiently reliable, this offers the potential for automatic measurement of the shapes of tidal features as well.

6.2 Training data

The sophistication of the CNNs used in this paper is limited by the size of the training data. The expert labels from A13 contain 305 tidal galaxies spread over 6 non-exclusive morphological classes of tidal feature. This places a fundamental limit on how much a convolutional network can generalize and learn to recognize such features. Pre-processing, shallow network design, augmentation, and dropout are all necessary to achieve our classification performance.

Larger training sets would provide constraining information to support CNNs with more free parameters. This in turn would allow for more complex predictions about the input images. In principle, a CNN could directly localize tidal features with bounding boxes (Huang et al. 2017), provide predictions for many different classes of tidal features (Simonyan & Zisserman 2014), and estimate tidal parameters like the length of a tidal tail (Toshev & Szegedy 2014). Our heatmap experiment (Section 6.1) provides compelling evidence for the plausibility of these applications if a sufficiently large training sample can be realized. We discuss three possibilities for this below.

Visually identifying large samples of galaxies with faint tidal structure is a daunting task given the relative rarity of such features at the typical surface brightness levels of current wide-field data sets. Most studies agree that to a surface brightness of $\mu \sim 26.5$–28 mag arcsec^{-2}, roughly 10–20 per cent of galaxies show evidence for faint tidal features (e.g. A13; Hood et al. 2018; Kado-Fong et al. 2018; Morales et al. 2018). In order to create a training sample of even $\sim$10 000 tidal systems, more than 100 000 galaxies would need to be visually inspected. Crowd-sourcing efforts like Galaxy Zoo (Lintott et al. 2008; Willett et al. 2013) could be an effective way to accomplish this but we note that tidal features from minor mergers and accretions are often rather subtle in appearance and visual identifications typically require some degree of interactive manipulation of pixel scaling and contrast. While citizen science may still prove an effective way forward, the accuracy of resulting tidal labels would need to be carefully verified, perhaps by checking against a smaller expert catalogue. Citizen science would also help mitigate the risk of a single expert producing classifications that systematically deviate from other experts.

Alternatively, or in conjunction, one could use synthetic training data from simulations. Individual tidal features can be simulated in exquisite detail (e.g. Johnston et al. 2008; Hendel & Johnston 2015) and large-scale hydrodynamical simulations of galaxy formation now have the resolution to resolve these features in populations of several thousand galaxies (e.g. Pop et al. 2017). With simulated data, mock observations could be made at many viewing angles and surface brightness thresholds in order to provide an arbitrarily large training sample. However, while simulations provide perfect information on tidal labels, they are unlikely to fully capture the development and evolution of real tidal features, impairing the ability of the classifier to detect such features.

Finally, transfer learning provides an indirect method to include training data. First, a convolutional network is trained to solve a related problem on an independent training set. The convolutional layers of the network become able to extract features relevant to that related problem. Secondly, those feature-extracting layers are used to construct a new convolutional network aimed at solving the target problem. The filters learned by those feature-extracting layers may be useful to re-apply. For example, learned filters that detect shapes and orientation on the related problem may be helpful for the target problem (see Yosinski et al. 2014). The features learned by CNNs trained on general galaxy morphology problems with far larger samples (Willett et al. 2013; Dieleman et al. 2015; Huertas-Company et al. 2015) could be particularly relevant for detecting faint tidal features. A recent application of this is provided by Ackermann et al. (2018), who use transfer learning in conjunction with CNNs to automatically identify images of galaxy mergers.

6.3 Application to new data

We ultimately aim to apply this method to detect tidal features in a large galaxy sample not previously classified. It is therefore important to ensure that that this method scales.

Each ensemble classifier makes tidal predictions on the order of 100 galaxies per second on a standard 2.4 Ghz CPU, or approximately eight million galaxies per CPU-day. This means that classifying forthcoming samples from LSST and *Euclid*, which will be several orders of magnitude larger than the A13 sample, is computationally feasible.

A13 manually removed images contaminated by stars, which would not be feasible for a large sample. However, automatic identification of contaminating stars is straightforward (Soumagnac 2015; Cabayol et al. 2018; Kennamer et al. 2018; Sevilla-Noarbe et al. 2018). Current methods reach an AUC score exceeding 0.99

on comparable CFHT imaging (Kim & Brunner 2017). For LSST-scale samples, we would use such methods to automatically remove contaminating stars prior to application of our convolutional neural network.

We removed as uninformative 8 per cent of images (136 of 1757) with expert labels of exactly 50 per cent confidence in tidal features. The performance metrics reported apply only to this slightly cleaner sample. Assuming classifiers guess randomly for such uncertain galaxies, and the true labels are equally random, the AUC scores of all the methods discussed would be slightly lower. This does not affect our demonstration of the relative strength of convolutional neural networks at detecting tidal features.

6.4 Potential bias

Scalability is only meaningful if we understand the biases involved in the classifications. There are two important sources of bias introduced by the classifier that need to be considered.

In the first case, the classifier may perform particularly poorly at recognizing some classes of tidal features (e.g. streams or shells). It is crucial to understand these biases so that they may be distinguished from genuine trends in the galaxy population. One way to approach this would be to construct a 'calibration' catalogue where the true tidal feature labels are known. This could be achieved through using multi-expert visual classifications, or even synthetic data. Given a calibration catalogue, one can measure how classifier performance varies for each tidal feature class. We measure the performance of our classifier by tidal debris class in Section 3.7. Should some classes be poorly recognized, one could either apply an appropriate correction or search for additional examples of that tidal feature class to improve performance.

On the other hand, within any given data set, bias may be triggered by the image context. Experts understand that they should not consider bright foreground or background objects, diffraction spikes or any other 'artefacts' when making a classification. CNNs have no such expertise unless inferred from the training data. Further domain-specific augmentations could help the classifier avoid confusion from these context biases. Adding synthetic observational effects would provide training examples to teach the classifier to ignore such effects and better handle, for example, classifications of galaxies in crowded images.

7 CONCLUSION

We have examined the performance of CNNs with dropout and augmentation to identify galaxies in the CFHTLS-Wide Survey that have faint tidal features in their outer regions. Learning the ideal features to extract from the pixel data and gradually increasing the pixel scale of feature maps make CNNs effective at classifying features in complex images. We have shown that appropriate preprocessing and augmentation combined with a relatively shallow network architecture is key to avoiding overfitting of the data. Randomized five-fold cross-validation verifies that our results are independent of which images are selected for training and which for testing. Training and testing five uniquely instantiated CNNs in two different ensemble configurations confirms that our results are statistically reliable and do not result from a fortuitous instantiation of initial weights. Through adding mock tidal features, we have shown that our method highlights image features that are found to be discriminatory without applying a parametric model.

Comparing the performance of our classifiers against previously published expert visual classifications, we find that our method achieves high (76 per cent) completeness and low (20 per cent) contamination. It also performs considerably better than other automated methods recently applied in the literature, namely the shape asymmetry method, a non-parametric approach developed for identifying post-merger galaxies by Pawlik et al. 2016), and WND-CHRM, a generic machine learning approach previously applied to image classification in astronomy (Shamir 2012).

Our demonstration of the effectiveness of CNNs represents a significant step forward in developing a fully automated method for faint tidal feature detection in galaxies. Indeed, most work in detecting and classifying tidal features in galaxies is still wholly or partially dependent on expert visual identification (e.g. Hood et al. 2018; Kado-Fong et al. 2018; Morales et al. 2018). This strategy is completely inadequate for the next generation of deep wide field surveys, such as LSST and *Euclid*, which will cover $\sim$15 000–20 000 deg^2 at unprecedented photometric depth (Laureijs et al. 2011; Robertson et al. 2017). While a limiting factor is the lack of currently available training data, the use of either citizen science labels, simulation data, or transfer learning are potential ways to address this. The development of a robust and efficient method to not only identify, but also characterize, faint tidal features around galaxies will enable the record of minor mergers and interactions to be mined in very large statistical samples. This will provide unique and previously inaccessible insight into the history of the galaxy population over cosmic time and facilitate the much-anticipated revolution that next generation facilities promise in terms of quantitative low surface brightness science.

ACKNOWLEDGEMENTS

We would like to thank the reviewer Lior Shamir for their helpful comments and suggestions.

MW acknowledges funding from the Science and Technology Funding Council (STFC) Grant Code ST/R505006/1. AMNF acknowledges support from STFC and the Alexander von Humboldt Foundation.

Based on observations obtained with MegaPrime/MegaCam, a joint project of CFHT and CEA/IRFU, at the Canada–France–Hawaii Telescope (CFHT) which is operated by the National Research Council (NRC) of Canada, the Institut National des Science de l'Univers of the Centre National de la Recherche Scientifique (CNRS) of France, and the University of Hawaii. This work is based in part on data products produced at Terapix available at the Canadian Astronomy Data Centre as part of the Canada–France–Hawaii Telescope Legacy Survey, a collaborative project of NRC and CNRS.

This research made use of the open-source PYTHON scientific computing ecosystem, including SCIPY (Jones et al. 2001), MATPLOTLIB (Hunter 2007), SCIKIT-LEARN (Pedregosa et al. 2011), SCIKIT-IMAGE (van der Walt et al. 2014), and PANDAS (McKinney 2010).

This research made use of ASTROPY, a community-developed core PYTHON package for Astronomy (The Astropy Collaboration 2013, 2018).

This research made use of the deep learning PYTHON package KERAS (Chollet et al. 2015), recently included within TENSORFLOW (Abadi et al. 2015).

All codes are publicly available on Github at www.github.com/mwalmsley/tidal_features_classifier (Walmsley 2018).

REFERENCES

Abadi M. et al., 2015, TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Available at: https://www.tensorflow.org/

Abadi M. G., Navarro J. F., Steinmetz M., Eke V. R., 2002, ApJ, 591, 499

Abazajian K. N. et al., 2009, ApJS, 182, 543

Ackermann S., Schawinski K., Zhang C., Weigel A. K., Turp M. D., 2018, MNRAS, 479, 415

Adams S. M., Zaritsky D., Sand D. J., Graham M. L., Bildfell C., Hoekstra H., Pritchet C., 2012, AJ, 144, 128

Atkinson A. M., Abraham R. G., Ferguson A. M. N., 2013, ApJ, 765, 28 (A13)

Ball N. M., Brunner R. J., 2009, Int. J. Mod. Phys. D, 19, 1049

Belokurov V. et al., 2006, ApJ, 642, L137

Bradley L. et al., 2018, astropy/photutils: v0.5. Available at: https://doi.org/10.5281/zenodo.1340699

Bridge C. R., Carlberg R. G., Sullivan M., 2010, ApJ, 709, 1067

Cabayol L. et al., 2018, preprint (arXiv:1806.08545)

Chollet F., 2016, Building Powerful Image Classification Models Using Very Little Data. Available at: https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html

Chollet F. et al., 2015, Keras. Available at: https://github.com/fchollet/keras

Conselice C. J., 2003, ApJS, 147, 1

Conselice C. J., 2014, ARA&A, 52, 291

Cooper A. P. et al., 2010, MNRAS, 406, 744

Cooper A. P., D'Souza R., Kauffmann G., Wang J., Boylan-Kolchin M., Guo Q., Frenk C. S., White S. D., 2013, MNRAS, 434, 3348

Darg D. W. et al., 2010, MNRAS, 401, 1552

Dieleman S., Willett K. W., Dambre J., 2015, MNRAS, 450, 1441

Duc P. A. et al., 2015, MNRAS, 446, 120

Erben T. et al., 2013, MNRAS, 433, 2545

Ferguson A. M. N., Mackey A. D., 2016, in Newberg H. J., Carlin J. L., eds, Astrophysics and Space Science Library, Vol. 420, Tidal Streams in the Local Group and Beyond. Springer International Publishing, New York, p. 191

Fitts A. et al., 2018, MNRAS, 479, 319

Fonseca P. G., Lopes H. D., 2017, preprint (arXiv:1710.08901)

Freeman P. E., Izbicki R., Lee A. B., Newman J. A., Conselice C. J., Koekemoer A. M., Lotz J. M., Mozena M., 2013, MNRAS, 434, 282

Gwyn S. D. J., 2012, AJ, 143

Hendel D., Johnston K. V., 2015, MNRAS, 454, 2472

Hood C. E., Kannappan S. J., Stark D. V., Dell'Antonio I. P., Moffett A. J., Eckert K. D., Norris M. A., Hendel D., 2018, ApJ, 857, 144

Hopkins P. F. et al., 2009, MNRAS, 397, 802

Hopkins P. F. et al., 2018, MNRAS, 480, 800

Huang J. et al., 2017, Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 3296

Huang J., Ling C. X., 2005, IEEE Trans. Knowl. Data Eng., 17, 299

Huertas-Company M., et al., 2015, ApJS, 221, 8

Hunter J. D., 2007, Comput. Sci. Eng., 9, 99

Johnston K. V., Hernquist L., Bolte M., 1996, ApJ, 465, 278

Johnston K. V., Bullock J. S., Sharma S., Font A., Robertson B. E., Leitner S. N., 2008, ApJ, 689, 936

Jones E. et al., 2001, SciPy: Open source scientific tools for Python. Available at: http://www.scipy.org/

Kado-Fong E. et al., 2018, ApJ, 866, 103

Kartaltepe J. S. et al., 2010, ApJ, 721, 98

Kennamer N., Kirkby D., Ihler A., Sánchez J., 2018, in Jennifer D., Krause A., eds, Proc. 35th Int. Conf. Mach. Learn. PMLR, Stockholm, p. 2582

Kim E. J., Brunner R. J., 2017, MNRAS, 464, 4463

Kormendy J., Kennicutt R. C., 2004, ARA&A, 42, 603

Lanusse F., Ma Q., Li N., Collett T. E., Li C. L., Ravanbakhsh S., Mandelbaum R., Póczos B., 2018, MNRAS, 473, 3895

Laureijs R. et al., 2011, preprint (arXiv:1110.3193)

LeCun Y. A., Bengio Y., Hinton G. E., 2015, Nature, 521, 436

Lee J., Yi S. K., 2017, ApJ, 836, 1

Lintott C. J. et al., 2008, MNRAS, 389, 1179

Lofthouse E. K., Kaviraj S., Conselice C. J., Mortlock A., Hartley W., 2017, MNRAS, 465, 2895

Lotz J. M. et al., 2008b, ApJ, 672, 177

Lotz J. M., Primack J., Madau P., 2004, AJ, 128, 163

Lotz J. M., Jonsson P., Cox T. J., Primack J. R., 2008a, MNRAS, 391, 1137

Lotz J. M., Jonsson P., Cox T. J., Croton D., Primack J. R., Somerville R. S., Stewart K., 2011, ApJ, 742, 103

Lu J., Behbood V., Hao P., Zuo H., Xue S., Zhang G., 2015, Knowl.-Based Syst., 80, 14

McKinney W., 2010, Data Structures for Statistical Computing in Python. Available at: http://conference.scipy.org/proceedings/scipy2010/mckinney.html

Malin D. F., Carter D., 1983, ApJ, 274, 534

Martin G., Kaviraj S., Devriendt J. E. G., Dubois Y., Laigle C., Pichon C., 2017, MNRAS, 54, 50

Martínez-Delgado D. et al., 2010, AJ, 140, 962

Miskolczi A., Bomans D. J., Dettmar R.-J., 2011, A&A, 536, A66

Morales G., Martínez-Delgado D., Grebel E. K., Cooper A. P., Javanmardi B., Miskolczi A., 2018, A&A, 614, A143

Oliphant T. E., 2006, A Bayesian perspective on estimating mean, variance, and standard deviation from data. Bringham Young University Scholars Archive

Orlov N., Shamir L., Macura T., Johnston J., Eckley D. M., Goldberg I. G., 2008, Pattern Recognit. Lett., 29, 1684

Pawlik M. M., Wild V., Walcher C. J., Johansson P. H., Villforth C., Rowlands K., Mendez-Abreu J., Hewlett T., 2016, MNRAS, 456, 3032

Pedregosa F. et al., 2011, J. Mach. Learn. Res., 12, 2825

Pedregosa F. et al., 2012, J. Mach. Learn. Res., 12, 2825

Petrillo C. E. et al., 2017, MNRAS, 472, 1129

Pop A.-R., Pillepich A., Amorisco N. C., Hernquist L., 2017, MNRAS, 480, 1715

Qu Y. et al., 2017, MNRAS, 464, 1659

Quinn P. J., 1984, ApJ, 279, 596

Robertson B. E. et al., 2017, preprint (arXiv:1708.01617)

Rodriguez-Gomez V. et al., 2016, MNRAS, 458, 2371

Russakovsky O. et al., 2015, Int. J. Comput. Vis., 115, 211

Sánchez H. D., Huertas-Company M., Bernardi M., Tuccillo D., Fischer J. L., 2017, MNRAS, 476, 3661

Schutter A., Shamir L., 2015, Astron. Comput., 12, 60

Sevilla-Noarbe I. et al., 2018, MNRAS, 481, 4, 5451

Shamir L., 2012, J. Comput. Sci., 3, 181

Sheen Y. K., Yi S. K., Ree C. H., Lee J., 2012, ApJS, 202, 8

Simonyan K., Zisserman A., 2015, International Conference on Learning Representations

Snyder G. F., Lotz J., Moody C., Peth M., Freeman P., Ceverino D., Primack J., Dekel A., 2015, MNRAS, 451, 4290

Soumagnac M., 2015, Doctoral thesis. UCL University College, London

Tal T., van Dokkum P. G., Nelan J., Bezanson R., 2009, AJ, 138, 1417

Tewari A., Bartlett P. L., 2005, J. Mach. Learn. Res., 8, 143

The Astropy Collaboration, 2013, A&A, 558, 53, 9

The Astropy Collaboration, 2018, AJ, 156, 123

Toomre A., Toomre J., 1972, ApJ, 178, 623

Toshev A., Szegedy C., 2014, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit, Curran Associates, Inc, New York. p. 1653

van Dokkum P. G., 2005, AJ, 130, 2647

van der Walt S., Schönberger J. L., Nunez-Iglesias J., Boulogne F., Warner J. D., Yager N., Gouillart E., Yu T., 2014, PeerJ, 2, e453

Walmsley M., 2018, Tidal Features Classifier: Initial public release - updated account. Available at: https://doi.org/10.5281/zenodo.1476538

White S. D. M., Frenk C. S., 1991, ApJ, 379, 52

Willett K. W. et al., 2013, MNRAS, 435, 2835

Yosinski J., Clune J., Bengio Y., Lipson H., 2014, in Ghahramani Z., Welling M., Cortes C., Lawrence N. D., Weinberger K. Q., eds, Advances in Neural Information Processing Systems 27, New York, Curran Associates, Inc., p. 3320

Zeiler M. D., Fergus R., 2014, Lecture Notes in Computer Science. Springer, Cham, p. 818

Zhang C., Ma Y., eds, 2012, Ensemble Machine Learning. Springer, Boston, available at: http://link.springer.com/10.1007/978-1-4419-9326-7

This paper has been typeset from a TEX/LATEX file prepared by the author.