Using transfer learning to detect galaxy mergers

Sandro Ackermann,1 Kevin Schawinski,1 Ce Zhang,2 Anna K. Weigel1 and M. Dennis Turp1

1 Institute for Particle Physics and Astrophysics, Department of Physics, ETH Zurich, Wolfgang-Pauli-Strasse 27, CH-8093, Zürich, Switzerland
2 Systems Group, Department of Computer Science, ETH Zurich, Universitätsstrasse 6, CH-8006, Zürich, Switzerland

ABSTRACT

We investigate the use of deep convolutional neural networks (deep CNNs) for automatic visual detection of galaxy mergers. Moreover, we investigate the use of transfer learning in conjunction with CNNs, by retraining networks first trained on pictures of everyday objects. We test the hypothesis that transfer learning is useful for improving classification performance for small training sets. This would make transfer learning useful for finding rare objects in astronomical imaging datasets. We find that these deep learning methods perform significantly better than current state-of-the-art merger detection methods based on nonparametric systems like CAS and GM20. Our method is end-to-end and robust to image noise and distortions; it can be applied directly without image preprocessing. We also find that transfer learning can act as a regulariser in some cases, leading to better overall classification accuracy ($p = 0.02$). Transfer learning on our full training set leads to a lowered error rate from 0.038$\pm$1 down to 0.032$\pm$1, a relative improvement of 15%. Finally, we perform a basic sanity-check by creating a merger sample with our method, and comparing with an already existing, manually created merger catalogue in terms of colour-mass distribution and stellar mass function.

Key words: methods: data analysis – techniques: image processing – galaxies: general

1 INTRODUCTION

Galaxy mergers are an important driver of the mass assembly and transformation of massive galaxies and the triggering of quasars (Toomre & Toomre 1972; Silk & Rees 1998; Sanders et al. 1988; Mihos & Hernquist 1996; Bell et al. 2003a; Springel et al. 2005; Hopkins et al. 2006; Treister et al. 2010, 2012; Weigel et al. 2017a). Several different methods have been previously used to detect galaxy mergers in observational data:

One way to detect mergers is the close-pairs method (e.g. Lin et al. (2004) or Woods & Geller (2007)). Here, luminosity peaks are algorithmically identified. Each one of these peaks is then considered as a target for collecting spectroscopic data. Close pairs of such spectroscopic targets in the image plane are considered as potential mergers. Redshift measurements can then be used to confirm that two candidates are close enough to be gravitationally interacting in a significant way. One problem with the close pairs method is imprecise radial distance measurements due to peculiar velocities. Furthermore, detecting the spectrographic targets requires a heuristic algorithm, which may be hard to hand-engineer, considering the diversity in the morphologies of mergers. It might even be impossible, for some merger images, to detect two separate luminosity peaks, as not every merger exhibits two distinct luminosity peaks.

Another approach purely relies on imaging data. Here, a handful of hand-crafted feature detectors are used for classification. Examples for this approach are the CAS features (concentration, asymmetry, clumpiness) by Conselice (2003) or GM20 (Gini coefficient and the second-order moment of the brightest 20% of the galaxy’s flux) by Lotz et al. (2004) and combinations and variants of those systems (Cotini et al. 2013; Hoyos et al. 2011; Goulding et al. 2017). Those non-parametric systems prescribe, in an algorithmic manner, how each feature should be extracted from a galaxy image; e.g. asymmetry from CAS is defined as the normalised residuals of comparing a flipped version of the galaxy image to the original image. Determining the individual values for each of the $n$ features allows the classification algorithm to distinguish between mergers and non-interacting galaxies, depending on where the analysed galaxy image lies in the $n$-dimensional feature space. A known problem with these non-parametric approaches using hand-crafted feature detectors is the difficulty of capturing the full diversity of merger ap-
Pearce, and the varying sensitivity of detection to each stage of the merger process, see the simulation studies by Lotz et al. (2008) and Lotz et al. (2010).

Until now, the accuracy of manual classification by human experts cannot be reached by automatic methods. The Galaxy Zoo (GZ)1 project byLintott et al. (2008, 2010) achieved visual morphological classification of around one million galaxies from the Sloan Digital Sky Survey (SDSS) by crowdsourcing the classification task out to citizen scientists on an online platform. Classifying mergers versus non-interacting galaxies was part of the first Galaxy Zoo run, and we will use those classifications later in this paper as our ground truth classifications to train our own classifier.

The problem of algorithmically categorising images into different classes is not a problem that is specific to merger detection in astronomy. Image classification is one of the main problems of the computer vision and machine learning community. They have developed, over the years, a wealth of methods to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.

We would like to emphasise that, when using deep CNNs, there are no hand-crafted feature detectors involved. The salient features for the specific classification task are discovered by the neural network automatically during training. Training is the process of tuning the free parameters of the neural network to solve image classification. Recently, with the advent of large labeled datasets and cheap computational resources, Convolutional Neural Networks (CNNs) have achieved a performance level that represents a significant improvement over more traditional computer vision methods (Krizhevsky et al. 2017). Today, the best CNN architectures can rival or even surpass the performance of human classifiers on some datasets (He et al. 2015). Deep CNNs have already been used successfully for regression and classification tasks with imaging data of galaxies (Dieleman et al. 2015; Hoyle 2016). It seems prudent to test how these state-of-the-art CNN architectures fare in our specific task of detecting mergers; this will be the main focus of this paper.
convex, we are resorting to gradient based methods (gradient descent and related algorithms) to do this optimisation step to arrive at $\hat{\theta}$. In theory, gradient descent can converge to any local minimum, and we have no guarantee of finding a solution that is at, or close to the global minimum. However, this does not seem to pose a very significant problem with currently employed neural network architectures (Goodfellow et al. 2014). Please note that there are alternative algorithms for non-convex optimisation, which handle local minima more gracefully, e.g. evolutionary algorithms for optimisation.

Having finished the training, we can use our function $f(\cdot | \hat{\theta})$ as an approximation of the ground truth function $f^*(\cdot)$. This second step with fixed weights is called inference.

A specific type of layer is the convolutional layer: Inspiration from the discovery of receptive fields of neurons in the visual cortex, and taking advantage of the translational symmetry of natural images lead researchers to experiment with this type of layer for solving visual processing tasks with ANNs (Fukushima 1980; LeCun et al. 1998; Krizhevsky et al. 2017). In a convolutional layer, each unit only receives input (i.e. has nonzero weights) from a local image patch of the precursor layer (receptive field). Many units (they constitute a so called feature map) in the convolutional layer share the same weight matrix, apart from two-dimensional translations in the image plane, so that each unit has its own receptive field. This is essentially the same as convolving the original layer with a convolution kernel corresponding to the weight matrix, and then using the activation function $\phi(\cdot)$ on each pixel. Doing this convolution for multiple different kernels yields multiple feature maps. The activations of those feature maps are essentially the activations of the convolutional layer and are the input to the next layer.

Convolutional layers exhibit translational invariance and they dramatically reduce the number of free, learnable parameters due to weight sharing and limited receptive fields. This speeds up the training process and acts as a strong regularizer. Convolutional layers are used extensively in today’s state-of-the-art DNNs for visual processing (see the seminal work by Krizhevsky et al. (2017)) and other tasks. These types of DNN architectures are called deep convolutional neural networks, or deep CNNs.

We will be using the Xception CNN architecture by Chollet (2016) for this paper.

### 2.2 Transfer Learning

The network architectures that are used in deep learning have a very high dimensional free parameter space, i.e. they have many different weights that need to be tuned during training. When trained on a limited number of samples (small training set), this gives the network enough capacity to fit to the noise or specific properties of that training set. This leads to a good accuracy on the training set, but the network may have a very high dimensional free parameter space, i.e. they can pre-train a SOTA classifier CNN on a the ImageNet dataset (Deng et al. 2009), which is a big dataset of a few million natural images from thousands of categories, containing categories like cars and dogs and cats and many others. A selection of some sample images including their class labels can be seen in figure 1. We then use the CNN with these pre-trained weights as the starting point for our merger classification training. We hypothesise that the initialisation with the pre-trained ImageNet weights can act as a regulariser. We will thus expect an improvement in the generalisation performance of the classifier to a level that is significantly above the level of just training the merger classifier from random weight initialization. This is one form of transfer learning.

### 2.3 Dataset

To train a classifier, we need a dataset which consists of pairs of images with the corresponding ground truth classifications (merger vs non-interacting). Our source for the imaging data is the Sloan Digital Sky Survey (SDSS) Data Release 7, where we use the SDSS online image cutout service\(^2\) to get RGB JPEG images of the galaxies of interest. For our ground truth classifications, we are using the crowdsourced labels from the Galaxy Zoo project, where we are interested in the weighted-merger-vote fraction $f_m$. We use the 3003 merger objects from the Darg et al. (2010) merger catalogue, which itself is based on $f_m$ from the GZ data. This catalogue takes all objects with $0.005 < z < 0.1$ and $f_m > 0.4$ and runs them through a second visual confirmation process, using human experts. This yields the 3003 merger objects in the catalogue. As our non-interacting galaxies sample, we choose 10000 GZ galaxies with $f_m < 0.2$ and in the same redshift range in a random draw. During training, we will do

\(^2\) http://skyserver.sdss.org/dr12/en/help/docs/api.aspx
2.4 Experiment

We are interested in two questions: How does a modern CNN architecture, trained on a merger dataset, compare to previous SOTA merger classifications (main experiment), and how does the classification performance of a CNN with transfer learning compare to a CNN with random initialization (lesion study). We hypothesise transfer learning to be useful especially in cases with small training set sizes, thus we will conduct the lesion study for different training set sizes to investigate the influence of training set size on the utility of transfer learning over random initialization. We will use artificially restricted training sets with training set sizes of \([300, 1500, 900, 500, 300]\) and test the superiority of a transfer learning approach for each one of them. For more technical details about the training refer to the corresponding parts of the appendix.

3 RESULTS

After training the CNN on the training set, we need to evaluate the performance of the classifier on the test set, which was never used during training. The trained classifier produces, for each galaxy image, an output \(p_m \in [0, 1]\), where a value of 0 means a classification as a non-interacting galaxy, and a value of 1 means a classification as a merger system. We chose a threshold of 0.5 for \(p_m \in (0, 1)\) to distinguish between the two categories and get a binary classification.

After obtaining the automatic classification for each image in the test set, we can quantify the performance of our method. We report precision, recall (or sensitivity) and \(F_1\) numbers of our method, and compare it to the performance of previous SOTA methods in Table 1. Precision \(p\), recall \(r\) and \(F_1\) are defined as follows:

\[
p = \frac{n_{TP}}{n_{TP} + n_{FP}}
\]

\[
r = \frac{n_p}{n_{TP} + n_{FN}}
\]

\[
F_1 = 2 \frac{p \cdot r}{p + r}
\]

Here, \(n_{TP}\), \(n_{FP}\) and \(n_{FN}\) refer to the number of true positive, false positive and false negative classifications respectively.

Table 1. Reporting precision, recall and \(F_1\) of our method, with a comparison to previous automatic visual classification methods. Keep in mind that only the recall \(r\) can be used as a valid comparison between the methods, as only this quantity is invariant under the different class ratios used for testing by the different authors.

\[
\begin{array}{cccc}
\text{method} & \text{recall } r & \text{precision } p & \text{ } F_1 \\
\hline
\text{Goulding et al. (2017)} & 0.75 & 0.90 & 0.82 \\
\text{Cotini et al. (2013)} & 0.8 & 0.8 & 0.8 \\
\text{Hoyos et al. (2011)} & 0.92 & 0.29 & 0.44 \\
\text{our method} & 0.96 & 0.97 & 0.97 \\
\end{array}
\]

stratified sampling from those two sets, so that each minibatch has the same number of images of merging galaxies and non-interacting galaxies.

In figure 2, we plot the test error rates of the CNN with transfer learning versus the test error rates of the CNN with random initialisation, for the different chosen training set sizes of \([300, 1500, 900, 500, 300]\).

Using 4-fold cross validation during training (cf. appendix), we obtained four independent samples of the test error, for every single experiment. We can use the one-tailed Welch’s t-test (Welch 1947) to calculate p-values for the null hypothesis of transfer learning having the same or higher mean test error as random initialisation.

A statistically significant \((p < 0.05)\) advantage of transfer learning over random initialisation can be observed for the training set sizes of 300 \((p = 0.05)\) and 3000 \((p = 0.03)\), a plausible advantage for the training set size of 1500 \((p = 0.12)\), and finally inconclusive results for the training set sizes of 900 \((p = 0.32)\) and 500 \((p = 0.72)\).

If we combine the different experiments using Stouffer’s Z-score method (Stouffer et al. 1949; Whitlock 2005), we arrive at a combined p-value of \(p = 0.02\), which leads us to the conclusion that transfer learning gives a significant advantage over random initialisation.

4 QUANTIFYING PERFORMANCE

4.1 Histogram and calibration plot

Classifying all images from the test set with our classifier gives us a \(p_m \in [0, 1]\) for every test set image (A value of 0 meaning classification as a non-interacting galaxy, and a value of 1 meaning classification as a merger system). We can immediately plot a histogram of those classifications (cf. figure 3). We can see that the CNN predominantly outputs \(p_m\) close to either zero or one, cases with unclear classification are rare.
Let us now focus on all galaxies that were classified into a bin between $p_m$ and $p_m + \Delta p_m$ for some $p_m$, and $\Delta p_m$ small. Then we can, knowing the true classifications in the test set, see if the fraction of true mergers in this bin is indeed close to $p_m$. This is called a calibration plot (cf. figure 4). We can see that the calibration is somewhat close to a diagonal from $(0,0)$ to $(1,1)$, i.e. if we randomly select $N$ galaxies closely around a certain $p_m$, then we can expect roughly $p_m \times N$ true mergers to be in that sample.

### 4.2 Recalibration

In order to further improve calibration, i.e. bring the data points in figure 4 to a diagonal from $(0,0)$ to $(1,1)$, we employ isotonic regression (Barlow et al. 1972; Chakravarti 1989). Isotonic regression is essentially the task of fitting a set of data points optimally with a monotonically non-decreasing model function. Here, we fit a monotonically non-decreasing function $p_{\text{true}} \approx c(p_{\text{classifier}})$ to the data points in figure 4. This allows us to transform any result directly taken from the classifier $p_{\text{classifier}}$, into an approximate true probability $p_{\text{true}}$. We used one third of the test set for the recalibration fit, and evaluated the quality on the remaining two thirds of the test set. For the calibration plot and histogram after recalibration, please refer to figures 6 and 5 respectively.

### 4.3 ROC (Receiver Operating Characteristic) curve

Let us repeat that the output of our classifier is continuous, $p_m \in [0,1]$. This means that, if we want a binary classification into either merger or non-interacting classes, we need to apply thresholding to the obtained $p_m$. The
Figure 6. Calibration plot after recalibration. The error bars in x-direction denote the limits of bins of \( p_m \), while the error bars in y-direction denote 2σ CIs. The calibration is very consistent with a diagonal crossing the origin and of slope 1, i.e. the classifier is well-calibrated. That means we can interpret \( p_m \) as a probability.

Figure 7. The ROC curve of the classifier. A curve close to the dotted diagonal would correspond to a classifier that just outputs a random \( p_m \) for each image. Our ROC curve is consistently and significantly above this diagonal, and reaches an AUROC (integral under the ROC curve) of 0.9922, which is very close to the maximum of 1. This tells us that our classifier performs well, even with high class skew (unbalanced classes) or at any a priori chosen threshold.

Figure 8. Examples (random draw) for true positive classifications in the test set, including \( p_m \). We typically see a high confidence \( p_m \approx 1 \). The classifier is able to correctly detect a wide variety of different merger morphologies.

ROC curve is a tool to quantify classifier performance without specifying a particular threshold a priori; we plot true positive rate (\( f_{TP} = \frac{n_{TP}}{n_{TP} + n_{FP}} \)) versus false positive rate (\( f_{FP} = \frac{n_{FP}}{n_{FP} + n_{TN}} \)), for all possible thresholds \( \in [0,1] \). The ROC curve is invariant under changes in class distribution (number of mergers versus number of non-interacting systems in the test set) (Fawcett 2006). This is useful in our case because we do not know the merger fraction a priori. The AUROC (Area Under ROC), the integral under the ROC curve, results in a single scalar to compare different classifiers; a value close to 1 means a close-to-perfect classifier. The AUROC also quantifies the probability that a randomly chosen merger image is classified with a higher \( p_m \) than a randomly chosen non-interacting system image (Fawcett 2006). In our case we found AUROC = 0.9922.

4.4 Failure modes

We provide some example images of true positive/negative and false positive/negative classifications in figures 8, 9, 10 and 11 respectively. Overall, the classifier does well with the expected diversity of merger system appearances, and it also seems to be able to correctly identify star overlaps. This level of generality might be quite hard to hand-engineer by adapting a system without feature learning like CAS. On the other hand, a significant part of the misclassifications are images that are also hard to correctly classify for human classifiers, like potential galaxy overlaps and very late and very early stage merger systems.
Transfer learning for merger detection

5 PROPERTIES OF THE MERGER SAMPLE

We created a merger sample by taking all the Galaxy Zoo I objects in the same redshift range as the Darg et al. (2010) merger catalogue, and then obtaining the classification \( p_m \) for each galaxy with our classifier. We used the classifier trained on the largest training set with transfer learning. Table 2 shows the top and bottom five objects according to \( p_m \).

Keep in mind that some of the Galaxy Zoo I objects were already in the dataset for training our classifier. However, using 4-fold cross validation during training, there is always at least one classifier, for every single object, that has not been exposed to this object during training. This means we can actually provide a \( p_m \) for every object without cheating by doing inference on the training or validation set.

To investigate the properties of this merger sample, we examine the distribution of detected mergers in colour-mass space and determine their stellar mass function.

Please note that here we are using a classifier, trained on a certain training set, for inference on a data set with with similar, but different underlying statistical properties. We do not have any guarantees that this will lead to a sensible merger sample. However, we will argue ex-post by comparing the resulting merger catalogue to the Darg et al. (2010) catalogue in terms of a few astrophysical quantities, and

Table 2. The top and bottom five objects in our merger sample, according to the mean of all \( p_m \). This table does not include any objects used during training. Notice that we have four different estimates for \( p_m \). This is due to using 4-fold cross validation during training, which gives us four different classifiers.

| objid          | ra        | dec       | \( P_m^{(0)} \) | \( P_m^{(1)} \) | \( P_m^{(2)} \) | \( P_m^{(3)} \) |
|---------------|-----------|-----------|----------------|----------------|----------------|----------------|
| 58772552819634248 | 10:56:39.17 | +67:10:49.0 | 1.000000       | 0.999997       | 0.999994       | 0.999996       |
| 587741816249516036 | 10:37:58.48 | +22:25:00.0 | 0.999996       | 1.000000       | 0.999981       | 0.999994       |
| 588011124116488195 | 13:15:35.06 | +62:07:28.6 | 0.999999       | 0.999970       | 1.000000       | 0.999997       |
| 587733080280465515 | 13:25:29.68 | +53:34:56.3 | 0.999996       | 0.999993       | 1.000000       | 0.999977       |
| 587742014375067679 | 15:07:55.82 | +17:21:50.9 | 0.999997       | 0.999973       | 0.999974       | 0.999999       |

Figure 9. Examples (random draw) for true negative classifications in the test set, including \( p_m \). We typically see a high confidence \( p_m \approx 0 \). The classifier seems to be able to correctly identify star overlaps and still classify the non-interacting galaxy as such.

Figure 10. Examples (random draw) for false negative classifications in the test set, including \( p_m \). The classifier is less confident for some of the examples, as is expected for a well calibrated classifier. Some false negatives seem to be part of very early or very late stage merger events.
finding reasonable agreement between the two catalogues in this regard.

5.1 Colour-Mass diagram

In Fig. 12 we show the distribution of merger galaxies that have been identified with the CNN ($p_m > 0.95$) in colour-mass space. Our (arbitrary) threshold $p_m > 0.95$ leaves us with 7980 objects (out of originally 328151). For the colour-mass diagram, shown on the left-hand side, we use a volume complete sample ($0.02 \leq z \leq 0.06$) to avoid a bias due to incompleteness. This selection effect can be avoided if the sample is split into mass bins. On the right-hand side, we thus illustrate the colour distribution of the entire sample in three different stellar mass bins. For comparison, we also show the properties of merger galaxies identified via visual classification (Darg et al. 2010) and of all SDSS galaxies within the same volumes. Note, that the apparent magnitudes used here have not been corrected for dust.

Fig. 12 illustrates, that major merger galaxies that have been identified via transfer learning lie within the same colour and mass range as visually classified mergers. Both samples span from the blue cloud, across the green valley, to the red sequence (Bell et al. 2003b; Baldry et al. 2004; Faber et al. 2007; Martin et al. 2007; Schawinski et al. 2014). Comparing the $u-r$ values of CNN and visually classified mergers in more detail shows that, compared to the Darg et al. (2010) sample, our sources tend towards redder colours. However, the Kolmogorov-Smirnov test (Eadie et al. 1971) shows, that this difference is only significant for the $9 < \log(M/M_\odot) \leq 10$ bin ($p$-value = 0.009). Hence only for this bin the two samples are likely to have been drawn from different distributions.

5.2 Stellar mass functions

In addition to the colour-mass diagram, we also determine the stellar mass function of our merger sample. Stellar mass functions are a sophisticated statistical measurement. Their determination includes correcting for selection effects and the resulting shape reflects the details of the true, underlying mass distribution.

Weigel et al. (2017b) use the Darg et al. (2010) catalogue to determine the stellar mass function of major merger galaxies (mass ratio up to 1:3) in the $0.02 < z < 0.06$ redshift range. They find that the space density of major merger galaxies is well fit by a single Schechter function with $\log(M^*/M_\odot) = 10.89 \pm 0.06$, $\log(\Phi^*/h^{3} Mpc^{-3}) = -3.43 \pm 0.02$, and $\alpha = -0.55 \pm 0.08$. We restrict our sample to the same redshift range, select galaxies for which $p_m > 0.95$, and use the same method (see Weigel et al. 2016) as Weigel et al. (2017b) to determine the stellar mass function of merger galaxies that have been identified with transfer learning.

Fig. 13 illustrates our results. We show the stellar mass function of merger galaxies that have been identified by the CNN in blue, the mass function of visually classified major mergers by Weigel et al. (2017b) in red, and the mass function of the entire galaxy sample in the same redshift range Weigel et al. (2016) in grey. Open markers (1/Vmax, Schmidt 1968), filled markers (SWML: Efstonthiou et al. 1988), and solid lines (STY: Sandage et al. 1979), illustrate the results of three independent mass function estimators. For the mass function of CNN identified mergers we find the following best-fitting parameters: $\log(M^*/M_\odot) = 10.81 \pm 0.04$, $\log(\Phi^*/h^{3} Mpc^{-3}) = -3.43 \pm 0.02$, and $\alpha = -0.62 \pm 0.09$. The shape of the mass function of mergers that have been identified via transfer learning thus resembles the mass function of visually classified major mergers. However, we find a significantly higher space density $\Phi^*$.

Using a different cut in terms of $p_m$ does not change $\Phi^*$. The normalisation, $\Phi^*$ does however increase with $p_m$. For $p_m = 0.90, 0.95, 0.99$ we determine $\log(\Phi^*/h^{3} Mpc^{-3}) = -3.36 \pm 0.02$; $-3.43 \pm 0.02$; $-3.53 \pm 0.02$, respectively.

To interpret the $\Phi^*$ difference in Fig. 13, it is important to be aware of the differences between our and the Weigel et al. (2017b) mass function. Weigel et al. (2017b) restrict their sample to major merger galaxies, i.e. they include a cut in terms of the mass ratio of the merging galaxies. The mass measurements are based on fits to the photometry (Darg et al. 2010). For each merging system, they include the mass of the more massive merging partner, if spectra are available. For both merging galaxies. They use the mass of the galaxy for which a spectrum is available, if only one of the merging galaxies has been observed spectroscopically. We do not include a mass ratio cut. Furthermore, we include all spectroscopically observed galaxies with $p_m > 0.95$ in our mass function. In contrast to the Weigel et al. (2017b) sample, we thus count systems double, if both merging partners have been observed spectroscopically and both have $p_m > 0.95$. Due to these differences in terms of sample selection, the $\Phi^*$
offset between our sample and the results by Weigel et al. (2017b) does not directly imply that, within the same volume, we are able to identify more merger galaxies with transfer learning than with a visual classification.

Fig. 13 illustrates another subtle difference between our sample of mergers that have been identified with transfer learning and the Darg et al. (2010) sample of visually selected mergers: whereas the Darg et al. (2010) sample is complete to log$(M/M_\odot) = 9$, the completeness of our sample only reaches to log$(M/M_\odot) = 10$. This is due to the difference in terms of colour, which we discussed in the previous section. Tending toward redder colours, our mergers exhibit lower mass-to-light ratios (low luminosity compared to their stellar mass) than the Darg et al. (2010) mergers. Mass-to-light ratios are used to translate a survey’s completeness in terms of luminosity into a completeness in terms of stellar mass (Pozzetti et al. 2010; Weigel et al. 2016). This conversion is particularly sensitive to the mass-to-light ratios of low mass, low redshift galaxies. The difference in terms of colour, which we illustrated in Fig. 12 and which was significant for the lowest mass bin, thus directly accounts for the difference in terms of completeness in Fig. 13.

Besides this difference in terms of mass-to-light ratios, Fig. 13 illustrates the overall consensus between visual and CNN based merger classifications.

6 CONCLUSIONS

We have shown that by using state-of-the-art CNNs, we can outperform the previous methods for automatic visual detection of galaxy mergers significantly. We also showed that for our dataset sizes, transfer learning by initialising with e.g. ImageNet weights can lead to a modest improvement in the generalisation power of the trained classifier. A sanity check of our method by creating a merger sample with our method and comparing the properties of this sample to the Darg et al. (2010) catalogue shows agreement in terms of colour-mass distribution and stellar mass function.

Our methods are not specific to merger classification and can be used for the general problem of detecting rare astronomical objects such as gravitational lenses (Marshall et al. 2016), galaxies with shocked interstellar medium (Alatalo et al. 2016), AGN ionization echoes (Keel et al. 2012) or ring galaxies (Buta 1995).

We would also like to emphasise the convenient property of our method to produce well-calibrated classifications, i.e. for each image, the classifier calculates a number $p_m \in [0,1]$, which can be interpreted as a true probability of the classified object being part of a merger system.

Please refer to the SpaceML\(^3\) project to access code,
full models of the classifiers, and a full table of the GZ I derived merger sample obtained from our classifiers.

ACKNOWLEDGEMENTS

KS and AKW acknowledge support from Swiss National Science Foundation Grants PP00P2_138979 and PP00P2_166159 and the ETH Zurich Department of Physics. CZ and the DSSLab gratefully acknowledge the support from the Swiss National Science Foundation NRP 75 407540_167266. IBM Zurich, Mercedes-Benz Research & Development North America, Oracle Labs, Swisscom, Zurich Insurance, Chinese Scholarship Council, the Department of Computer Science at ETH Zurich, and the cloud computation resources from Microsoft Azure for Research award program.

REFERENCES

Alatalo K., et al., 2016, ApJS, 224, 38
Baldry I. K., Glazebrook K., Brinkmann J., Ivezić Z., Lupton R. H., Nichol R. C., Szalay A. S., 2004, The Astrophysical Journal, 600, 681
Barlow R., Bartholomew D., Bremner J., Brink H., 1972, Statistical Inference Under Order Restrictions: The Theory and Application of Isotonic Regression., Wiley, New York
Bell E. F., McIntosh D. H., Katz N., Weinberg M. D., 2003a, ApJS, 149, 289
Bell E. F., McIntosh D. H., Katz N., Weinberg M. D., 2003b, The Astrophysical Journal Letters, 585, L117
Buta R., 1995, ApJS, 96, 9
Chakravarti N., 1989, Mathematics of operations research, 14, 303
Chollet F., 2016, Xception: Deep Learning with Depthwise Separable Convolutions (arXiv:1610.02357)
Cichy R. M., Khosla A., Pantazis D., Torralba A., Oliva A., 2016, Scientific Reports, 6
Conselice C. J., 2003, The Astrophysical Journal Supplement Series, 147, 1
Gotz S., Ripamonti E., Cacciapuoti A., Colpi M., Ceca R. D., Magelli M., Severgnini P., Segreto A., 2013, Monthly Notices of the Royal Astronomical Society, 431, 2661
Cybenko G., 1989, Mathematics of Control, Signals, and Systems, 2, 303
Darg D. W., et al., 2010, Monthly Notices of the Royal Astronomical Society, 401, 1043
Daugman J. G., 1985, Journal of the Optical Society of America A, 2, 1160
Deng J., Dong W., Socher R., Li L.-J., Li K., Fei-Fei L., 2009, in 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, doi:10.1109/cvpr.2009.5206848, https://doi.org/10.1109/cvpr.2009.5206848
Dieleman S., Willett K. W., Dambre J., 2015, Monthly Notices of the Royal Astronomical Society, 450, 1441
Eadie W. T., Drijard D., James F. E., 1971, Amsterdam: North-Holland, 1971
Ellis G. G., Ellis R. S., Peterson B. A., 1988, Monthly Notices of the Royal Astronomical Society, 232, 431
Faber S., et al., 2007, The Astrophysical Journal, 665, 265
Fawcett T., 2006, Pattern Recognition Letters, 27, 861
Fukushima K., 1980, Biological Cybernetics, 36, 193
Goodfellow I. J., Vinyals O., Saxe A. M., 2014, preprint, (arXiv:1412.6544)
Goodfellow I., Bengio Y., Courville A., 2016, Deep Learning. MIT Press
Goulding A. D., et al., 2017, Publications of the Astronomical Society of Japan, 70
He K., Zhang X., Ren S., Sun J., 2015, in 2015 IEEE International Conference on Computer Vision (ICCV). IEEE, doi:10.1109/iccv.2015.123, https://doi.org/10.1109/iccv.2015.123
Hebb D., 1949, The Organization of Behavior: A Neuropsychological Theory, Wiley
Hopkins P. F., Hernquist L., Cox T. J., Di Matteo T., Robertson B., Springel V., 2006, ApJS, 163, 1
Hornik K., 1991, Neural Networks, 4, 251
Hoyle B., 2016, Astronomy and Computing, 16, 34
Hoyos C., et al., 2011, Monthly Notices of the Royal Astronomical Society, 419, 2703
Jones J. P., Palmer L. A., 1987, Journal of neurophysiology, 58, 1233
Keel W. C., et al., 2012, MNRAS, 420, 878
Krizevsky A., Sutskever I., Hinton G. E., 2017, Communications of the ACM, 60, 84
Lecun Y., Bottou L., Bengio Y., Haffner P., 1998, Proceedings of the IEEE, 86, 2278
Lin L., et al., 2004, The Astrophysical Journal, 617, L9
Lintott C. J., et al., 2008, Monthly Notices of the Royal Astronomical Society, 389, 1179
Lintott C., et al., 2010, Monthly Notices of the Royal Astronomical Society, 410, 166
Lotz J. M., Primack J., Madlau P., 2004, The Astronomical Journal, 128, 163
Lotz J. M., Jonsson P., Cox T. J., Primack J. R., 2008, Monthly Notices of the Royal Astronomical Society, 391, 1137

Figure 13. Stellar mass function of mergers that have been identified with transfer learning in comparison to the mass functions of all galaxies and of visually classified major mergers. In blue, we show the stellar mass function of galaxies with $p_m > 0.95$ in the $0.02 < z < 0.06$ range, which we determine by using the method by Weigel et al. (2016). We compare our results to the mass function of visually selected major mergers (red, Weigel et al. (2017b)) and of all galaxies Weigel et al. (2016) in the same redshift range. Open markers, filled markers, and solid lines show the results of three independent mass function estimators (see text and Weigel et al. (2016) for more details).
Transfer learning for merger detection

A1 Main Experiment

- **Hypothesis:** CNNs (with transfer learning) are able to outperform state-of-the-art techniques for merger classification.
- **Proxy:** We measure the quality of our approach in terms of precision, recall and F-1 score at a classification threshold of $p = 0.5$. We also evaluate the ROC curve and measure the AUROC.
- **Protocol:** We conduct $K$-fold cross validation in the following way: For each of the $K$ iterations, we use one fold as the validation set, one fold as the test set, and the rest of the folds as the training set. First, we replace the last fully connected layers, trained on the the original ImageNet dataset, with two fully connected layers (random initialisation) with only two outputs (corresponding to our two-class classification task). We train just those FC weights for 40 epochs with the rest of the layers frozen. We then use SGD with momentum 0.9 and a learning rate of $0.000015$ and use the validation set accuracy for early stopping. We report the quality scores on the test set from the cross validation loop. This leaves us with $K$ samples for each one of our quality scores.

- **Expected Result:** The classification performance of CNNs with transfer learning dominates state-of-the-art methods, according to the chosen metrics.

A2 Lesion Study: The Impact of Transfer Learning

- **Hypothesis:** Transfer learning outperforms deep learning with random initialisation.
- **Proxy:** We compare the different outcomes in term of test error.
- **Protocol:** We first generate subsets of the dataset with sizes of 10%, 17%, 30%, 50% and 100% of the full dataset. For each of these datasets, we run $K$-fold cross validation, once using transfer learning and once using random initialisation. This results in $5 \times 2 = 10$ independent $K$-fold cross validation experiments. We use the same protocol for transfer learning as in the main experiment above. For deep learning with random initialisation, we first randomly initialise the weights of the network and then use SGD with momentum 0.9 and a learning rate of 0.000015 for training. We use the validation set accuracy for early stopping. We report the quality scores on the test set from the cross validation loop. This leaves us with $K$ samples of the test error, for each one of the ten cross validation runs.

- **Expected Result:** Comparing the quality scores for the two approaches, one should observe that the quality of transfer learning is better compared to random initialisation, especially for small training set sizes.

We choose $K$ by taking into account both available data and available computational resources. We end up choosing $K = 4$.

During training, we re-balance the data using stratified sampling for each mini-batch; i.e. each mini-batch contains the same number of images of merger and non-interacting systems.
This paper has been typeset from a T\TeX/\LaTeX\ file prepared by the author.