האוניברסיטה העברית בירושלים
THE HEBREW UNIVERSITY OF JERUSALEM

Detection of Galaxy Mergers and their Stages using Machine Learning

זיהוי מיזוגי גלקסיות והשלבים שלהם בעזרת מערכות לומדות

The Hebrew University of Jerusalem, Faculty of Mathematics and Natural Sciences, Racah Institute of Physics

האוניברסיטה העברית בירושלים, הפקולטה למתמטיקה ולמדעי הטבע, המכון לפיסיקה

Final thesis for Master's degree in Natural Sciences

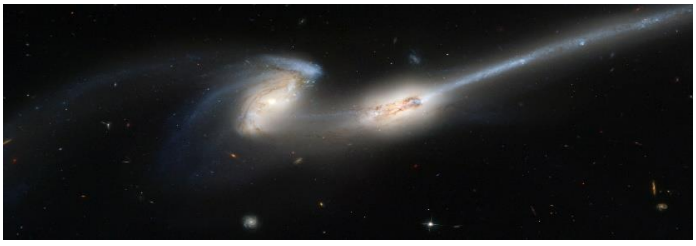
עבודת גמר לתואר מוסמך במדעי הטבע

Submitted by Alon Haviv – 203145420

Thesis Instructor – Professor Avishai Dekel

November 28, 2021

כ"ד בכסלו התשפ"ב



Abstract

Galaxy mergers are one of the most important processes of galaxy evolution. Yet, the merger rate at different epochs and the timescales of their stages of evolution are still not fully understood, mostly because of the difficulty in detecting them without bias. We use a Machine Learning based Convolutional Neural Network (CNN) technology to develop an automated tool that can distinguish between images of merging galaxies from non-merging galaxies out to a look-back time of 10 Gyr ($z = 2$), and separate their internal phases, without human biasing. We train our tool on images from a hydro-gravitational cosmological simulation (Horizon-AGN), where we know the ground truth for each galaxy and achieve > 90% accuracy, and then apply it to Hubble-Space-Telescope (HST) images of real galaxies from the CANDELS survey (GOODS-S and GOODS-S). We measure time in terms of the natural timescale for the merger process, which is expected to be the dynamical time of the dark-matter halos at the given epoch, about 15% of the cosmological Hubble time at that epoch. We find that the best observational time-scale of the entire major-merger event is 0.6 ± 0.1 dynamical times, and for the following 3 main phases: **Pre-Merger** (approaching galaxies), **In-Merger** (from the first-passage till final coalescence) and **Post-Merger** (from the final coalescence till relaxation), it is: 0.3 ± 0.05, $0.15^{+0.05}_{-0.025}$ and 0.15 ± 0.05 dynamical times respectively. The duration of the **Pre-Merger** phase was set to match the combined duration of the other 2 phases, which were discovered by the CNN. We are the first to estimate the duration of the different stages of the merger. We measure the merger-rate per galaxy per Gyr as: $MR(z) = (0.011 \pm 0.009) \cdot (1+z)^{3.55 \pm 0.91}\ Gyr^{-1}$. The merger-rate matches an earlier work that used a different simulation and different observational data, which serves as a validation for our technique and for the robustness of the results.

הקדמה

מיזוג גלקסיות הוא אחד מהתהליכים החשובים ביותר בהתפתחות גלקסיות. אולם קצב המיזוגים בתקופות שונות והזמנים האופייניים לשלבי ההתפתחות שלהם עודנם לא מובנים כיאות, בעיקר בשל הקושי לזהותם באופן אובייקטיבי. אנו משתמשים בטכנולוגית Convolutional Neural Network (CNN) מבוססת מערכות לומדות, על מנת לפתח כלי אוטומטי שיוכל להבחין בין תמונות של גלקסיות מתמזגות לכאלו שלא ב-10 מיליארד השנים האחרונות ($z = 2$), ולהפריד בין השלבים הפנימיים שלהם, ללא הטיות אנוש. את הכלי אנו מאמנים בעזרת תמונות מהסימולציה הקוסמולוגית והידרו-כבידתית (Horizon-AGN), שבה אנו יודעים את "האמת" עבור כל גלקסיה ומשיגים דיוק של למעלה מ-90%, ואז מפעילים אותו על תמונות של גלקסיות אמיתיות מתצפיות CANDELS של טלסקופ החלל "האבל" (HST) (GOODS-S ו-GOODS-N). אנו מודדים זמנים במונחים של סקאלת הזמנים הטבעית של תהליך המיזוג, שאמורה להיות הזמן הדינאמי של הילת החומר האפל בהתאם לתקופה, כ-15% מזמן האבל באותה תקופה. אנו מוצאים שהזמן האופייני התצפיתי המתאים ביותר לתהליך המיזוג כולו הוא 0.6 ± 0.1 זמנים דינאמיים, ולכל אחד מ-3 השלבים הבאים: Pre-Merger (התקרבות הגלקסיות), In-Mergeer (מהמעבר הראשון להתלכדות הסופית) ו- Post-Merger (מההתלכדות הסופית לרגיעה), הזמנים הם 0.3 ± 0.05, $0.15^{+0.05}_{-0.025}$ ו- 0.15 ± 0.05 זמנים דינאמיים בהתאמה. משך שלב ה Pre-Merger נקבע כך שיתאים למשך הזמן הכולל של 2 השלבים האחרים, אשר התגלו ע"י ה-CNN. אנו הראשונים שחקרו את הזמנים האופייניים של השלבים השונים של המיזוג. אנו מודדים את קצב המיזוגים לגלקסיה למיליארד שנים כ: $MR(z) = (0.011 \pm 0.009) \cdot (1+z)^{3.55 \pm 0.91}\ Gyr^{-1}$. התוצאה הזו מתאימה לתוצאות מעבודה קודמת אשר עשתה שימוש בסימולציה אחרת ומידע תצפיתי שונה, ובכך משמשת כאישרור לשיטה שלנו ולאמינות התוצאות.

Table of Contents

1. Introduction

Galaxy mergers are events in which 2 or more galaxies fuse together to form a larger galaxy. As such, it is an important part of a galaxy formation and evolution (Mo et al. 2010). In the standard ΛCDM model, as the universe evolves and space expands, galaxies are formed and evolve as well, in a hierarchical way: Either by gas accretion or by merging galaxies into one another (Duncan et al. 2019, Almeida et al. 2014). When gas is being accreted into the galaxy, it is done either directly or by forming clumps that accrete more gas or merge into larger clumps, and then fall and merge into a galaxy. When galaxies merge, it can either be a "minor-merger" if the larger galaxy is more than 4 times the mass of the smaller one, or a "major-merger" if the mass ratio is less than 4. And the galaxies become larger and larger the more they merge. Hence the hierarchical growth.

Galaxy mergers, especially major-mergers, are very dramatic events. They cause dramatic morphological distortions to the galaxies, can change the global shape of a galaxy from a thin, cold disc into a hot, elliptical and as clouds of gas collide, they can trigger high bursts of star formation (high SFR). However, not every aspect of these events is well understood. For an instance, the time-scales of the event or its subphases (from the approaching, through the merging and to the post-merging), the specific-star formation rate (sSFR) and the merger rate themselves in different redshifts.

Part of the reason for the difficulties comes from the fact that in observations we see only a single snapshot of the event and not the whole process, and another part comes from the difficulty in detecting those mergers (Conselice 2006; Lotz et al. 2008; Conselice 2014; Man et al. 2016). There are 2 main methods for detecting galaxy-mergers (Conselice et al. 2003, Lotz et al. 2004): One is by finding close pairs and measuring the distance between 2 galaxies from the angular separation and the redshift difference. If the distance is smaller than a threshold, which depends on the masses of the galaxies, then the galaxies are labeled as a merger. Another method is by recognizing galaxies with disturbed morphologies, which is a strong suggestion (but not solely) for galaxy merging and interactions (Mundy et al. 2017; Duncan et al. 2019). Both methods are biased. The close-pairs method is biased towards pre-mergers only, and doesn't consider the velocity vectors of the galaxies, which can change the results. The disturbed morphology method aims to the most disturbed cases and may miss galaxies that show less disturbances, such as early pre-mergers, or include galaxies that do show such features but aren't mergers, like clumpy galaxies due to SFR or accretion. To summarize, there is a need for a more general and unbiased tool that can use all kind of features in order to detect galaxy mergers. Such a tool is Machine Learning.

1.1. The main phases of a Major-Merger

We divide the major-merger event into 3 main phases:

1. "Pre-Merger" (**preM**) – Starts with the approaching of the progenitors, roughly when their distance is 4 × radius of the main progenitor, and ends at the time of first closest approach, namely the first pericenter, or first passage.
2. "In-Merger" (**inM**) – Starts at the first pericenter and ends at the final coalescence.
3. "Post-Merger" (**postM**) – Starts at the final coalescence and ends when the galaxy is relaxed, meaning it doesn't show serious distorted morphology.

- Galaxies that are in any of these phases are considered as "Mergers" (**merger**).
- Galaxies that are relaxed or haven't merged for a long enough time (a longer time than the averaged relaxation time) are called "No-Mergers" (**no-merger**).

One of our goals is to find the time-scales for each of these phases. In the **preM** phase we expect to see 2 (or more) progenitors with significant morphological distortions only near the end when they are very close to each other. The **inM** phase is characterized by very extreme distortions, bursts of star formation and usually we can still see 2 cores. The **postM** phase is expected to have 1 core, but with very visible tidal features and distorted morphology. In the **no-merger** phase we expect to see "calm" galaxies, with an undisturbed symmetric morphology and low SFR. Figure **1** shows a schematic diagram of the major-merger event and its phases.

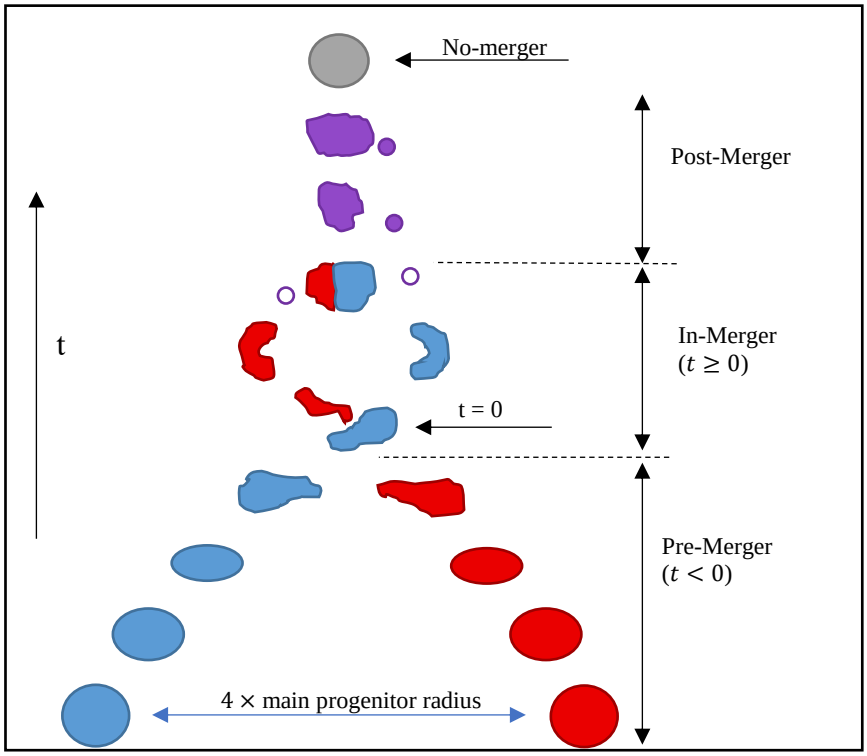


***Figure 1**: A schematic description of the merger event. There're 3 main phases: Pre-Merger, In-Merger and Post-Merger. $t = 0$ is the first passage and the beginning of the In-Merger phase.*

1.2. The dynamical-time scaling

We wish to derivate a redshift-depended dynamical time scale that we can use in order to describe the duration of the phases in different redshifts, with a unitless time parameter (t/t_{dyn}). The Pre-Merger phase is basically a free-fall of the 2 progenitors and the In-Merger + Post-Merger phases end when the resulted galaxy relaxes. In both cases, the bigger the scales of the galaxies and their dark matter halos (mass, size and distance), the longer these processes are expected to last on average. For a particular galaxy, other variables such as the impact parameter, cooling efficiency or differences in the structure of the galaxy may have an impact on the time-scales, but on average we expect them to cancel out and be marginally depended on the scales of the whole system. We explained before that these scales change by redshift as the universe expands. Therefore, instead of defining the observational time windows of these phases with absolute time units (Gys), which need to vary with redshift, we can normalize them by a redshift-depended time scale of a gravitational process of the entire system (linked to the mass, size and distance scales of the system), and get a unitless time-parameter that can define fixed observational time windows for the merger phases. This redshift-depended time-scale is the "Dark-Matter Halo's dynamical time" and by defining the observational time windows for the phases as fractions of the dynamical time, we get redshift-depended time-scales that can be expressed with fixed, unitless numbers. Example: $t/t_{dyn} = 0.5$ for all redshifts if both t and t_{dyn} change by redshift in the same way.

What is the "Dark-Matter Halo's dynamical time"? According to Dekel & Birnboim (2006) this is the crossing time of a dark matter halo in a virial equilibrium in a given redshift:

$$t_{dyn} \equiv \frac{R_{vir}}{V_{vir}}, \quad V_{vir}^2 = \frac{GM_{vir}}{R_{vir}} \qquad (1)$$

M_{vir}, R_{vir} and V_{vir} are the system's characteristic mass, radius and velocity at the virial equilibrium. It yields:

$$t_{dyn}(z) = 2.5\ Gys \cdot (1+z)^{-3/2} \cdot \left[\Delta_{200}(z) \cdot \Omega_{m,0}/0.3 \cdot h_{0.7}^2\right]^{-0.5} \qquad (2)$$

Where:

$$\Delta_{200}(z) \cong \left(18\pi^2 - 82\Omega_\Lambda(z) - 39\Omega_\Lambda^2(z)\right)/\Omega_m(z)$$

$$\Omega_m(z) = \Omega_{m,0}(1+z)^3\left(\Omega_{\Lambda,0} + \Omega_{m,0}(1+z)^3\right)^{-1}$$

$$h_{0.7} = H_0/70\ km\ s^{-1}\ Mpc^{-1}$$

And for a flat universe: $\Omega_\Lambda(z) = 1 - \Omega_m(z)$

Figure 2 shows the values of $t_{dyn}(z)$ with the cosmology: $\Omega_{\Lambda,0} = 0.7$, $\Omega_{m,0} = 0.3$:

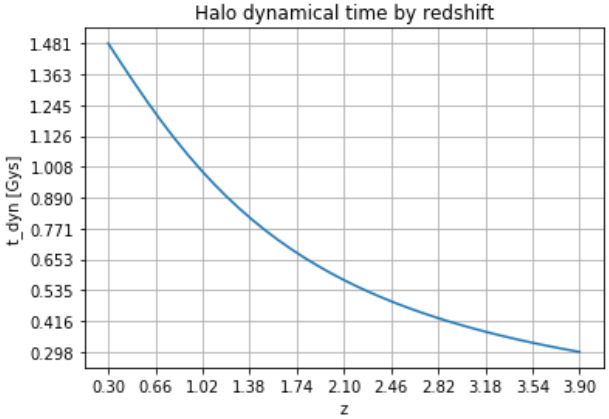


Figure 2: *Values of the dark-matter halo dynamical time over redshift, in Giga years.*

1.3. Merger-Fraction and Merger-Rate as functions of redshift

If we wish to understand the importance of mergers in the galaxy evolution, a main objective is to evaluate the merger rate in different redshifts. If we take a sample of galaxies we can check what are the **merger-fraction** among them within redshift bins, meaning the fraction of all galaxies at a given redshift that undergo a major-merger (mass ratio < 4), namely are in one of the merger stages as defined in section 1.1. Various of studies estimated the merger-fraction to be a power-law with respect to (1+z) (Press & Schechter 1974, Bond et al. 1991, Lacey & Cole 1993, Neistein & Dekel 2008a,b , Duncan et al. 2019, Ferreira et al. 2020). The main justification for the power-law behavior is the strong decrease of the galaxy density and of the gravitational pull between galaxies, as the universe expands. Because the gravitational pull behaves as an inverse power law of the scale-factor a ($F_G \propto a^{-2}$) and the density as a^{-3}, the percentage of galaxies that merge decreases as a power law. There are of course other factors such as the increase of the mass of each galaxy which contributes to a higher gravitational pull, but since it depends on accretion and mergers as well, meaning on the gravitational pull between objects, and therefore again on the scale-factor, its effect may change the strength of the power law, but otherwise maintain the same basic functionality. The scale factor is an inverse power law of (1+z). Therefore, the merger-fraction is given by:

$$MF(z) = f_0 \cdot (1+z)^{\alpha} \qquad (3)$$

f_0 and α are constants that should be found. Notice that if we change the merger's duration (τ_{obs}) it changes the number of galaxies that are counted as mergers and therefore changes the merger-fraction.

The **merger-rate** is the probability for a galaxy at redshift z to be detected during a major merger (its duration is τ_{obs}), per time unit (Gyr). It is given by dividing the merger-fraction by the merger duration τ_{obs} (in Gys) in each redshift. By this we get a measure that is independent of the duration assumed for the merger. The merger-rate is therefor:

$$MR(z) = \frac{MF(z)}{\tau_{obs}} = R_0 \cdot (1+z)^{\beta}\ Gyr^{-1} \qquad (4)$$

R_0 and β are constants that should be found. Dekel et al. 2013 estimates the specific-accretion rate of a halo, which includes all the mass that falls into the halo, at $z > 1$ to be:

$$\frac{\dot{M}_h}{M_h} = 0.030\ (1+z)^{2.5}\ Gyr^{-1} \qquad (5)$$

Since halo-merging is a subcase of accretion with the same underlying physics, it can be used as a theoretical estimation for the galaxy merger-rate with the slope $\beta \approx 2.5$. However, because the quantity of the accretion is all the mass that falls in, the normalization may not be accurate for our specific case of major-mergers of high-mass galaxies ($M_* \geq 10^{10} M_{\odot}$). Furthermore, because **halo-merger** and **galaxy-merger** are not always the same (there could be several galaxies within a single halo, especially at earlier times), even the power law may be somewhat different.

Neistein & Dekel 2008 also estimates the exact halo merger rate for different mass-ratios and final halo-masses (figure 6 in Neistein & Dekel 2008). For example, for halo mergers with final mass $10^{12} M_{\odot}$ and mass ratio $\mu > 0.3$ and $z{\sim}2$ we get:

$$MR_{N\&D} \approx 0.45\ Gyr^{-1} \qquad (6)$$

It can be a theoretical estimation for the magnitude of the merger-rate at a given redshift, which gives the normalization factor, but with the same caveat of the halo-merger not being exactly the same as galaxy-merger.

1.4. Machine Learning as a tool for detecting galaxy mergers

These days we can access very large databases of galaxies. Large surveys like the Sloan Digital Sky Survey (SDSS) (York et al., 2000) or the Cosmic Assembly Near-Infrared Deep Extragalactic Survey (CANDELS) (Grogin et al., 2011, Koekemoer et al., 2011) mapped thousands of galaxies in the sky, which is too much for the manual classification methods. In addition, today there are also huge cosmological simulations, such as Illustris and Horizon-AGN, that mimic the evolution of the universe and the formation of galaxies on a large scale ($\sim 100h^{-1}Mpc$ and more). The combination of large amount of data (in our case images observed through different filters) and simulations that allow us to track the histories of galaxies, build their merger-trees and get the "ground truth" for the mergers, make it ideal for Machine-Learning techniques.

Machine-Learning, and specifically Deep Learning, is a technology that developed a lot in the recent years and is able to deal with large amount of data, by letting a computer program to learn independently the best criteria for classification tasks, such as the one we are dealing with here. One of the advantages of "unsupervised" Deep Learning approaches, such as Convolutional Neural Networks (CNNs), which is a type of Deep Learning that is specifically for computer vision problems, is that they can use all the information in an image and not just a pre-defined list of parameters, and by that find features that we never considered or are very hard for us to quantify. Our strategy is to use CNNs, train them on mocked images generated from the Horizon-AGN cosmological simulation with the same characteristics of real galaxy images from CANDELS, and then use the trained networks on CANDELS to detect major-mergers. We also plan to use this tool to detect the different phases of the mergers, their and the entire event's time-scales, and the merger-rate as a function of redshift.

It is worth mentioning that CNNs were used recently for astronomical problems, for example for galaxy morphological classification, segmentation and deblending (Huertas-Company et al. 2018; Reiman & G¨ohre 2019; Huertas-Company et al. 2019; Cheng et al. 2019; Martin et al. 2019). It was even used for galaxy mergers detections, for example by Ackermann et al. (2018) who used an SDSS catalog classified by eye, and Ferreira et al. 2020 who trained a network on IllustrisTNG simulation and used it on different CANDELS fields, but used a constant observational time scale in all redshifts. It is worth mentioning that, to the extent of our knowledge, none of them tried to detect the major-mergers with a redshift-depended observational time-window that scales as the halo's dynamical time, and also constrain the time-scales of the inner phases, like we do in this work.

1.5. Our goals

In this thesis we'll train and test CNNs on mock-images from the Horizon-AGN simulation, in various cases, and apply them to real images from CANDELS. The goal is to detect galaxy major-mergers and their phases, estimate their durations, calculate the merger-rate as a function of redshift and to do it all with unitless time-scales, by normalizing the absolute time by the DM halo's dynamical time at each redshift. Since the work of Ferreira et al. (2020) is the most similar to ours (trained on IllustrisTNG and applied to CANDELS), we will compare our results with theirs. They in turn compared their results with Kartaltepe et al. (2015) and Duncan et al. (2019).

We start in section 4.1, by reproducing the results from Ferreira et al. (2020) by classifying **mergers** vs **no-mergers**, **preM** vs (**inM** + **postM**) and calculating the merger-fraction and merger-rate, all in CANDELS. It serves as a validation to our techniques. In section 4.2 we find an estimation for the duration of the (**inM** + **postM**) phase and the entire major-merger event, from the simulation. In section 4.3 we detect and estimate the duration of the inner "In-Merger" phase, from the simulation. In section 4.4 we reclassify **mergers** vs **no-mergers**, **preM** vs (**inM** + **postM**) and recalculate the merger-fraction and merger-rate, all in CANDELS but this time with our redshift-depended time-scales.

2. Data

Our goal is to use a Machine Learning technique that we train and test on a simulation's mock-images, and apply it to real observations. For the real observations catalog we used HST-CANDELS, while the simulated mock-images catalogs were generated by the Huertas-Company from the Horizon-AGN hydrodynamical cosmological simulation, and given to us. In the following sections we will explain how we got these catalogs and how they were generated.

2.1. HST-CANDELS

The **Cosmic Assembly Near-infrared Deep Extragalactic Legacy Survey** (**CANDELS**) is the largest project in the history of the **Hubble Space Telescope (HST)**, with 902 assigned orbits (about 60 continuous days) of observing time. Hence the name: HST-CANDELS. It was carried out between the years 2010 and 2013 and the goal of the survey was to explore galactic evolution in the early universe and obtain near infrared and visible images with the Wide Field Camera 3 (WFC3) and Advanced Camera for Surveys (ACS). Our catalog of real CANDELS images was generated by taking stamps of 128×128 pixels around known galaxies in the 2 CANDELS field: GOODS-S and GOODS-N, in 7 filter bands: F160W, F125W, F105W, F850LP, F775W, F606W and F435W. 9,420 galaxies in total. After filtering out low-mass galaxies with stellar mass less than $10^{10} M_{\odot}$, we were left with 1,063 galaxies in redshift $0.5 \leq z \leq 3$ and just 573 galaxies in redshift $0.5 \leq z \leq 2$.

2.2. The Horizon-AGN simulation

Horizon-AGN is a cosmological N-body + hydrodynamical simulation (Dubois et al., 2014, 2016), that has a comoving box size of $L_{box} = 100h^{-1}Mpc$, contains 1024^3 collision-less Dark-Matter particles (N-body interactions), and that was run considering initial conditions drawn from the WMAP-7 cosmology (Komatsu et al., 2011). The simulation employs the adaptive mesh refinement Eulerian hydrodynamics code – RAMSES (Teyssier, 2002), and the initially coarse 1024^3 grid is adaptively refined, in a quasi-Lagrangian manner, down to a spatial resolution of 1 proper kpc and temporal resolution of 17M years. Among the physical processes that are required to obtain realistic galaxies and that are implemented in this simulation: Cooling and heating of gas, star formation, black holes formation, and feedback from stellar winds, supernovae and Active Galactic Nuclei (AGNs).

2.2.1. Generating synthetic catalogs from a simulation

As we mentioned before, we got our simulated images from Huertas-Company, who used the Horizon-AGN simulation to generate catalogs of synthetic images of galaxy mergers and no-mergers with the same characteristics as in the CANDELS images. They did it by constructing merger trees for all sorts of galaxies and fed different snapshots to a code that generates synthetic images, as if they were taken by CANDELS cameras. The process is as following:

1) In the simulation, detect galaxies at redshift z = 0 using the AdaptaHOP structure finder algorithm (Aubert et al., 2004), with at least 50 stellar particles. It leads to galaxies with a minimum stellar mass of $10^8 M_{\odot}$.

2) Then run the simulation backwards and when a galaxy finally splits (recognized as 2 galaxies by the AdaptaHOP algorithm), with a stellar mass ratio $\geq 1{:}\,4$ between the 2 progenitors, label the last snapshot before the split as "The Merger" snapshot, marked by $t = 0$. When running time forwards, this snapshot represents the first pericenter (or first passage) of a merger event. This builds merger trees for all the galaxies found at redshift 0. The stellar mass ratio of at least 1:4 means we only consider major-mergers.
3) The beginning of the major-merger event is taken as the snapshot in which the distance between the 2 progenitors equals 4 times the radius of the main progenitor, where the radius is defined as the average of the three semi-axes that are obtained after fitting an ellipsoid to the distribution of stellar particles of the galaxy. The end of the major-merger event is taken symmetrically, with an equal number of snapshots before and after "The Merger" snapshot. All snapshots prior to "The Merger" ($t = 0$) are declared as "Pre-Mergers" with a negative time mark, and all later snapshots are declared "Post-Mergers" with a positive time mark.
   Note: Here "Post-Mergers" includes the "In-Merger" phase. Only later we have divided it into the 2 phases: **inM** and **postM**.
4) Then several filters are applied, such as: If one of the progenitors was only formed too close to the merger or if the stellar mass of one of the progenitors is less than $10^{10} M_{\odot}$ at the first snapshot, then the entire merger event is discarded. After this step we have a catalog of 273 "major-merger" events with their snapshots.
5) If a galaxy is isolated, meaning it wasn't interacting with other galaxies within $\pm 1 Gys$, it is added into the "isolated" catalog of isolated galaxies. Again, a filter of $M_* \geq 10^{10} M_{\odot}$ is applied.
6) For each snapshot of a galaxy in the "major-mergers" or "isolated" catalogs, a cubic volume is taken around the galaxy, with an edge length of 8 times the radius of the galaxy (the radius as defined above). This volume should encapsulate all, or at least most of the stellar particles of the galaxy and its progenitor-companion, in case of a pre-merger.
7) A radiative transfer code called "SUNSET" is then used to simulate the emission of these stellar particles in the observed frame in the 7 HST-CANDELS filters: The 3 NIR bands of the WFC3 (F160W, F125W, F105W) and the 4 visible bands of ACS-WFC (F850LP, F775W, F606W, F435W). It produces 7 images for each axis (X, Y, Z) for each snapshot. SUNSET considers parameters such as the age of the stellar particles, their metallicity models, initial mass, redshift and the instrument we want to mimic (WFC3 or ACS). The physical size of the pixels = 0.06 arcsec.
8) The images are then converted to the HST-CANDELS system by applying the same zero-points for the fluxes, according to the formula:

$$F_{CANDELS} = F_{SUNSET} \cdot 10^{0.4(ZP_{CANDELS} - ZP_{SUNSET})} \qquad (7)$$

   Where F_{SUNSET} is the image's flux produced by SUNSET, $F_{CANDELS}$ is the re-scaled flux in the HST-CANDELS system, ZP_{SUNSET} is SUNSET's zero-point and it equals to 48.6 for every band and $ZP_{CANDELS}$ is CANDELS's zero points and corresponds to the values group of values displayed in the last column (mag) of Table 1.
9) Then each image is convolved with the proper Point-Spread-Function (PSF) of the relevant instrument and band filter from CANDELS, and cropped to a size of 128×128 pixels. Since the physical size of a pixel in ACS is 0.03 arcsec, and we want all the images to have the same size (0.06 arcsec, as of WFC3), the PSF of the relevant bands had to be rescaled to match the

new size. At the end of this step we have complete catalogs of synthetic mock-images of the galaxies, as if they were observed in different bands using the Advanced Camera for Surveys (ACS) and the Wide Field Camera 3 (WFC3) of CANDELS. But without noise or background objects.

Notes:

- In some cases in this work we refer to "early" or "late" snapshots from the "major-mergers" catalog. By that we simply mean snapshots of an earlier or later time *t*, regardless of the phase.
- Real backgrounds were added to the images from samples of CANDELS fields during the training and testing processes of the images. It allowed us to get both the HST noise and background objects (such as other galaxies that are not part of the merger). And by sampling different backgrounds for the same images during different training stages, it also adds to the variety of the data and its effective size, which is important in Machine Learning.
- As we will see, the decision to define the duration of the merger events by the time when the distance between the 2 progenitors is 4 times the radius of the main progenitor (the merger's end time is symmetrical), and the decision to take as no-mergers only isolated galaxies that didn't, and won't interact with other galaxies for $\pm 1 Gys$, is a bit arbitrary. Part of this work, specifically section 4.2, is dedicated for choosing optimal time-cuts and if necessary mix galaxies from both catalogs.
- Due to a small and unrepresentative number of images in high redshifts ($z > 2$), and because during the network training we had to keep a balanced, equal number of images from each redshift range, we had decided to use only images of redshifts $0.5 \leq z \leq 2$ (the lowest redshift in all the catalogs, including CANDELS, is $z = 0.5$).

Table **1** shows the main parameters of the selected HST-CANDELS filters, and Figure **3** shows the wavelength coverage and associated colors of the filters.

Instrument	Filter	Mean Wavelength [Å]	Zero Point [Jy]	Zero Point [mag]
WFC3	F160W	15392.3	1138.1	25.940
WFC3	F125W	12516.2	1564.3	26.230
WFC3	F105W	10584.2	1975.2	26.268
ACS	F850LP	9082.3	2237.8	24.862
ACS	F775W	7729.7	2513.6	25.654
ACS	F606W	6034.9	3232.2	26.480
ACS	F435W	4348.0	4040.8	25.670

***Table 1**: Main parameters of selected HST-CANDELS bands.*

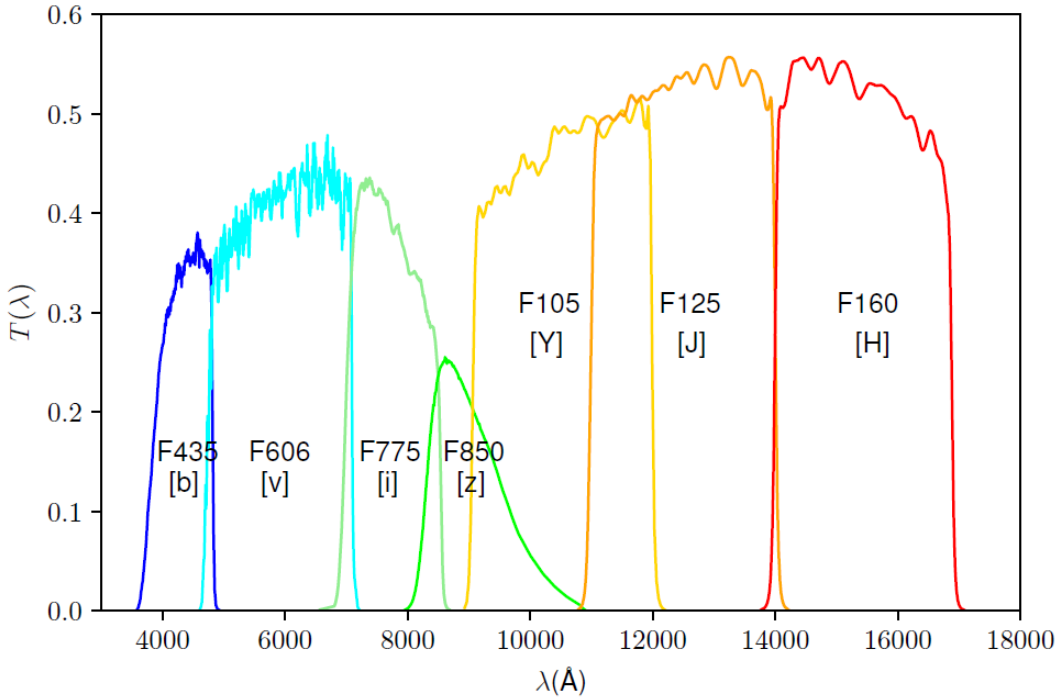


Figure 3: *Wavelength coverage of HST-CANDELS filters.*

2.2.2. Setting the backgrounds

In order to generate mock-images that faithfully mimic real observations, we have to add to them noise samples and background objects that might enter the frame, that characterize CANDELS images. The background objects are especially important if we want to train a network that can detect mergers from no-mergers, by looking at the morphological features and not only counting the number of objects in the frame. In order to do so, real backgrounds were added to the images from CANDELS fields during the training and testing processes of the images. It allowed us to get both the HST noise and background objects (such as other galaxies that are not part of the merger). And by sampling different backgrounds for the same images during different training stages, it also adds to the variety of the data and its effective size, which is important in Machine Learning. The challenge of course, is to sample backgrounds that on one hand have background objects, just like we see in real CANDELS images, and on the other hand don't have objects that overlap with the main galaxy/ies in the main image. And it should be so in all the filters.

For each image, the process of sampling the background is done by the following steps:

1) Sample a random point within the CANDELS fields. Calculate the physical coordinate of the point and apply it to all the filters, the get the same location in the sky.
2) For each filter take a 128×128 pixels stamp around the point.
3) For each filter stamp, perform a pixel-wise multiplication of the stamp with the image, and for each resulted image calculate the *total_flux* (sum of all the pixels' flux) and the *max_flux_pixel* (highest pixel value) and the ratio:

$$\beta \equiv \frac{total\_flux}{max\_flux\_pixel} \quad (8)$$

4) If $\beta < 4.4$ is satisfied **in all the filters**, then the stamp is approved as a background for that image and directly added (in all the filters). If $\beta \geq 4.4$ for at least one filter, then the stamp is discarded and repeat step (1).

The idea of equation (8) is that for any of the filters, if there's an overlap between an object in the background stamp (which is significantly brighter than the noise around it) and an object in the main image, then the multiplication results with many pixels of high S/N in the overlapping area. Areas with no such overlapping are expected to have a very low S/N value after the multiplication. If there is such overlapping, then the sum of signals from the bright pixels in the overlapped area will be significantly higher than the maximal pixel value, which is most likely one of them. Hence high β. If there isn't such overlapping, then there's a low S/N and the sum of the pixels is at the scale of the sum over the noise, and the brightest pixel is likely to be just a random multiplication of a single bright noise pixel with one of the actual objects (could be high). Hence low β.

The specific threshold of $\beta < 4.4$ was calibrated by testing 100 random images, 50 from the "major-mergers" catalog and 50 from the "isolated" catalog, with an arbitrary "rich" background stamp for each (meaning we chose stamps with objects and not just empty noise). We aimed for a threshold value for β that minimalized $2FP + FN$ rates (FP = Falsely approved stamps, FN = Falsely rejected stamps). The factor 2 is because we wanted to give a higher weight for FP, because it's much worse to use a bad background stamp with objects that overlap the galaxy/ies in the main image, than to use a too quiet stamp without as many background objects as we would have wanted. The value of 4.4 successfully minimalized the sum $2FP + FN$ with the rates $FP = 4\%, FN = 8\%$.

In appendix (1) we show an example of the technique with a good background and a bad background. We also show there a sample of several mock-images next to real images, for impression.

3. Machine Learning

Machine Learning technology became in the past couple of years a very useful tool to work with big data and to find correlations and patterns in the data. In this project we use a Machine Learning to classify between different types of galaxies (**mergers**, **no-mergers** and inner phases **preM**, **inM** and **postM**), by their morphological features, and from it draw important conclusions such as the merger-rate and merger durations as a function of redshift. Machine Learning technology is essentially a computer program that makes conclusions or decisions over data, and improves its performance with experience (exposure to training datasets). The process of "improving with experience" is called "learning" and the conclusions that the program draws are called "predictions". The performances are measured by comparing the predictions to a known set of true answers/labels, quantified by a pre-defined error-function. Note that it is up to us to define what exactly are those "conclusions or decisions" that the machine should make. For example, in our case it is classifications between classes of galaxies (**mergers**, **no-mergers**…) while on other cases it could be "Regression", which is assigning continues values (like mass) to the data.

In practice, we first have to train/teach a "model" – a specific realization of Machine Learning program, and then to test it before we make an actual use of it. For this reason, we take 2 data-sets of galaxy images that were already labeled by hand: A training-set and a testing-set. The bigger the data-sets the better the performances are going to be. The data-sets are drawn from the simulated catalogs: "major-mergers" and "isolated", which is why we know the true label of each image ("Merger", "No-Merger" or any other phase). The data-sets should be drawn from the same distribution, but must not include the same images, so we select images for each one by random with the rule that different snapshots of the same galaxy **cannot** be in more than one set. After that we can use the trained and tested model on real images from the CANDELS catalog. It is also a common practice to use a third data-set – a "validation" set, which is serves for an automated tuning of the very design and architecture of the model (often called "Hyper-parameters Tuning"). In this work we have many cases of already small data-sets, which is why we decided to calibrate the models manually beforehand, and thus divide the data to only 2 data-sets (training and testing) instead of 3. More of it in part (3.1.3).

3.1. Deep Learning with Convolutional Neural Network (CNN)

There are many different approaches in Machine Learning. We use Deep Learning with Convolutional Neural Networks (CNNs), which is a state-of-the-art tool for solving computer vision problems (Goodfellow et al. 2016) that is gaining popularity among galaxy merger studies (Ackermann et al. 2018; Pearson et al. 2019; Bottrell et al. 2019). Deep Learning techniques are able to perform unsupervised feature extraction from a given data. In Deep Learning, the data is often processed in a multistage manner, where each stage, also called "layer", searches for different types of features in the data. Together the layers and the ways they're connected form a "network", and the exact design of the network and its layers is called "the network's architecture". In a CNN, convolutional layers use convolution operations on multidimensional data, such as images, to extract features that can then be used for classification tasks in regular fully connected layers at the top of the CNN architecture. We will explain it:

3.1.1. How does it work?

A convolutional layer has 3 main parameters: The *number of filters* (**not** the band-filters that we mentioned before!), the *kernel-size* and the *activation function*. Here a filter is a small matrix of a single feature (a simple example is a line, or a corner), the *kernel-size* is the filter size and the *activation function* is used to control/normalize the result of the filter being applied to an input image. When a filter is applied to a part of the input image, the result is a number that tells the likelihood of the filter's feature to appear in that location in the image. The value is then fed as an input to the *activation function* which amplifies or suppresses it according to the desired behavior. In the convolution process of a full convolutional layer, each filter scans the input image, resulting in a reduced matrix where each element describes the likelihood of the filter's feature to appear (after applying the *activation function* of course). The result is a series of reduced matrices of feature-locations (and if we use multi channels/bands then each value is a vector of the size of the number of channels). Each matrix is then used as the input image of the next convolutional layer, which acts the same but this time looks for combinations of features, in order to find more complex patterns and features. Pooling operations are also used and usually located between blocks of convolutional layers with the goal of changing the input image to a lower (or higher) resolution, by down-sampling only specific elements (with maximal values for example). In the CNN terminology, a filter is also called a "neuron" and the connections between the neurons symbolize what filters are used for more complex combinations in the next layer. Dropout layer, which randomly turns off a percentage of the features given by the dropout-rate, is used to prevent the network from overfitting by strongly relay on a specific trajectory of neurons. Usually, a CNN network has several blocks of convolutional layers, with several convolutional layers in each, and pooling and dropout layers at the desired locations throughout the network. At the end of the journey, we get many small matrices in a multidimensional structure, which describe all sorts of complex patterns that the network found. A Flatten layer is then applied to collect it all into a 1D vector of high-level features, and then fully-connected layers down-sample the desired ones according to another activation function, up until we have only a few elements left. At the end the number of these elements is the number of classes we wish to classify (two classes if we do **mergers** vs **no-mergers**), and their values are the probabilities of the original input image to match each class.

3.1.2. The training process

For the above to work successfully, the filters need to be set to the most relevant features and sub-features. In other words, we must have the "right" matrices. The process that does it is the "training" (or "learning") and it requires a representative data-set, the "training-set", with known "*true labels*":

- All the filters are initially set with arbitrary matrices.
- We scan the training-set several times where each time is called an "*epoch*". During each *epoch* we slice (differently) the training-set to *batches* of equal "*batch-size"*.
- To increase the variety of the images we then perform <u>random</u> augmentations over each image:
    - Anti-clock rotation between 0 – 90 degrees.
    - Horizontal shift of up to 15% of the image's width.
    - Vertical shift of up to 15% of the image's height.
    - Horizontal flip.
    - Vertical flip.
    - Black-off three boxes of edge-size of 10% of the image's size at random locations.

- Add a random background from CANDELS field for each image in the *batch*, as described in section 2.2.2.
- The network then runs over the full *batch*, trying to make predictions. The predicted labels and the *true-labels* are fed to an error function, which compares them and outputs an error value.
- The network then uses an *optimizer-function* that updates the filter matrices according to the performance on the *batch* and the learning-rate, which is a parameter that sets the step-size/scale of each update. Usually, an *optimizer-function* uses a form of gradient-descent method, to converge on the filter-matrices values that give the minimal error-value. The learning-rate is important here because a too large one will cause drastic changes in the filter matrices, which can miss the minimal point, while a too small one will case too small changes that might get stack in a local minimum instead of the global one.
- The process continues for all the *batches* of all the *epochs*.

Another thing that is crucial for the training is to have a large, balanced training-set. The large size of the training-set is important in order to let the network learns a variety of images and make general conclusions about which features are important, while a too small data-set might either cause the network to set its filters specifically for the few images in the set (simply memorize it by heart), or failing outright. A balanced set is also important in order to prevent the network from setting up filters that favor one class (the bigger one) over the other (the smaller one), simply because that is what leads to a smaller error during the training. In this project we needed to balance the data-sets in 2 ways:

1. **Balance by phase** – It means that when we classify **mergers** vs **no-mergers** we have an equal number of merger images and no-merger images, and among the merger images there are equal numbers of **preM** and (**inM** + **postM**) images. When we classify **preM** vs (**inM** + **postM**) or **inM** vs **postM** etc, we have equal numbers of images of each class/phase. This is the most important balancing as it directly affects the error-function by the explanation above.
2. **Balance by redshift** – We sort all the images into redshift bins: 0.5-1, 1-1.5, 1.5-2, and make sure we have an equal or near equal number of images within each bin. This is the secondly important balancing.

In all the cases the balancing is done by down-sampling images randomly until a balance is reached.

<u>A note about data sizes in high redshifts:</u>

In the results section 4 we distinguish between **mergers** and **no-mergers** while the **no-mergers** class is composed of a mix of isolated galaxies from the "isolated" catalog, and "late" galaxies ($t/dynT \geq 0.5$) from the "major-mergers" catalog. Because in the simulated catalogs in redshifts $z > 2$ we have only a few hundreds of images, and among them only a few dozen are "late" ($t/dynT \geq 0.5$), and they also need to be divided into several datasets (training and testing), then we don't have representative data for this redshift range that we can train the machine by it. Furthermore, because we need to balance the data by redshift bins, which means that we should take an equal number of images in lower redshift and higher redshift bins, even if there are many more images in low redshifts, then by using $z > 2$ we end up with too small and unrepresentative data sets for all the redshifts, which we cannot train the machine by it (and we have tried). For this reason, in this project we took only galaxies of redshifts $0.5 \leq z \leq 2$ and we allowed a non-perfect balancing between redshift bins, up to 20%

difference between the bin sizes (this number was selected because it still allowed us to reproduce results done by other works in section 4.1.

3.1.3. Calibrations and final architecture

The performance of a CNN network depends on its exact architecture, the learning-rate, number of *epochs*, *batch-size*, what filters we use and more. Therefore, it is important to "tune" or "calibrate" these "hyperparameters" correctly before we can use the network. We did it manually by 2 series of tests followed by an attempt to reproduce results done by another paper (Ferreira et al. 2020): The first is Major-Mergers vs Isolated, which is the easier case where each class includes images from the relevant catalog "major-mergers" or "isolated", respectively. The second series is the stress-tests of Pre-Mergers vs Post-Mergers, all from the entire "major-mergers" catalog. In each series we checked different values and combinations of the following hyperparameters:

- The number of convolutional layers within a convolutional block.
- The number of convolutional layer blocks.
- The filter-matrix sizes of the convolutional layers (features).
- Amount and locations of dropout layers.
- The best combination of band-filters (colors).
- Learning rate.

Although we cannot say that we've tested all possible combinations, we did converge on a set of hyperparameters that we later on used to test if we can reproduce the results done by Ferreira et al. 2020 (see results in section 4.1). The success gave us the final confirmation of our calibrations.

While we used the same calibrated hyperparameters from above in all the cases we studied in section 4, the number of *epochs* and the *batch*-size were recalibrated manually and specifically for each of the cases, by performing a series of several runs with different values until we got the best performance. The reason is that the number of *epochs* and the *batch*-size are highly sensitive to the sizes of the data-sets, while the network architecture is sensitive to the specific task, be it classifying galaxies or other objects like numbers. The reason for it is that the number of *epochs* and the *batch-size* determine the variety of input that the network gets in each training step ("*batch"*) and in total during the whole process, which is obviously strongly limited by the total size of the data, while the architecture determine the features that the network is looking for, which in other words is: "What are we trying to classify?". Those features don't change very much between different runs of similar cases, for example when we classify 2 phases with different time-cuts, such as "In-Merger" vs "Post-Merger" (section 4.3), or even when we shift to different phases. Also remember that the calibration series included 2 different cases, and the same goes for the reproducing of the results of other works.

The final architecture and hyperparameters that we used:

- 3 convolutional blocks, with a varying number of feature-filters: 16, 32 and 64, and a varying filter-size: 3, 5 and 5.
- Each convolutional block contains 2 identical *convolutional* layers with "Relu" *activation-function*, and a *max-pooling* layer of size 2×2 with a stride of 2.
- A *dropout* layer with *dropout-rate* of 0.2.
- A fully connected layer with 512 units and "Relu" *activation-function*.
- The final fully connected layer with 2 units and "Softmax" *activation-function*.
- The *loss function* is "Categorical Crossentropy".
- The *optimizer function* is "Adam".
- We used the following band-filters: F105, F160, F435, F606 and F850.
- The number of *epochs* and the *batch-size* were calibrated specifically before each case.

3.2. Metric

In binary classification problems it is common to name the two classes as "*Positives*" (*P*) and "*Negatives*" (*N*). We can then describe the full confusion matrix with the following terminology:

TP = True Positives
TN = True Negatives
FP = False Positives
FN = False Negatives

	Predicted *N*	Predicted *P*
Real *N*	***TN***	***FP***
Real *P*	***FN***	***TP***

Table 2: *An example of a binary confusion matrix.*

Throughout this work we will measure the performance of the networks by 2 interchangeable binary metrices:

Metric 1:

- Accuracy (Acc): The percentage of correct predictions.
- Balance-rate (BR): *1* – The difference-rate between the 2 classes.

Metric 2:

- Recall (Rec): The percentage of correct "*Positives*".
- Precision (Prc): The purity of correct "Positives".

The Metrices' parameters can be mathematically expressed as follows:

$$\begin{cases} \text{Rec} & = \dfrac{TP}{TP + FN} \\ \text{Prc} & = \dfrac{TP}{TP + FP} \end{cases} \quad (9)$$

$$\begin{cases} \text{Acc} & = \dfrac{TP + TN}{2} \\ \text{BR} & = 1 - (TP - TN) = 1 - TP + TN \end{cases} \quad (10)$$

In a balanced case where $TP + FN = TN + FP = 1$ the 2 metrices are interchangeable in the following way:

$$\begin{cases} \mathrm{Rec} &= \mathrm{Acc} + \frac{1}{2}(1 - \mathrm{BR}) \\ \mathrm{Prc} &= \frac{\mathrm{Acc} + \frac{1}{2}(1 - \mathrm{BR})}{2 - \mathrm{BR}} \end{cases} \quad (11)$$

$$\begin{cases} \mathrm{Acc} &= \frac{\mathrm{Rec} \cdot (2 \cdot \mathrm{Prc} - 1)}{2 \cdot \mathrm{Prc}} + \frac{1}{2} \\ \mathrm{BR} &= 2 - \frac{\mathrm{Rec}}{\mathrm{Prc}} \end{cases} \quad (12)$$

* Note that $\mathrm{Prc} = 0 \; iff \; TP = 0$ and then also $\mathrm{Rec} = 0$ for any value of *TN*, and Acc and BR are not well defined in equation *(12)*.

3.3. Error Measurements

Throughout this work we use the results of our technique to draw conclusions about the universe. Particularly in parts 4.2.3 and 4.3 we estimate the observational duration of phases of the major-merger event, by comparing the metric's scores over the testing-set with different observational time windows. It is therefore important to estimate the error range of our results.

There are 2 types of errors: One is the epistemic error, that comes from uncertainty in the very technology that we use, and the other is the aleatoric error, that comes from variations in the data, such as noise etc.

The epistemic error comes from the uncertainty in the model itself for different cases, and to what extent we can trust the choice of the hyperparameters, the architecture and Machine Learning technology as a whole. It is possible that a different and more sophisticated technology would give us different results, especially in the cases studied in parts 4.2.3 and 4.3. This error is very difficult and even impossible to estimate. There are some attempts in the literature to estimate some aspects of it for very specific cases, such as the "Monte Carlo dropout" technique (Cook et al. 2000; Huertas-Company et al. 2019) that estimates the uncertainty in the training process of specifically a Bayesian neural network, which is a different architecture than what we used and is not relevant for us. Instead, in appendix 3 we try to attack it from a different angle by putting other results into a test – checking what can we get if we do try a **merger** vs **no-merger** classification with other time-cut results (from part 4.2.3). It is not perfect because we still rely there on machine learning technology, but since the **merger** vs **no-merger** classification case is relatively more trustworthy (simply because it matches other papers) than the classification cases we ran in parts 4.2.3 and 4.3 (because they are unique for us), it gives us some measure of certainty in the results. In the discussion (5) we suggest more tests that can help us constrain the error estimation.

The other type of error, the aleatoric error, comes from variations in the data, noise and backgrounds and to what extent the training and testing datasets are similarly representative. This type of uncertainty we do estimate using the following receipt:

1) In each case, when we test our network over a testing set, we repeat the testing 50 times while resampling the backgrounds anew randomly at each time.
2) We then calculate the averaged predictions-vector and averaged accuracy. It is supposed to cancel the effect of noise and background biasing from a specific background sampling.
3) We calculate the mean-prediction-error of every image prediction using the averaged accuracy ($\overline{acc}$) and the size of the testing-set – *n*, with the following formula:

$$\text{mean-prediction-error} = \pm\sqrt{\overline{acc}\cdot\frac{1-\overline{acc}}{n}} \qquad (13)$$

4) We cast the averaged predictions to the binary classes and from it calculate the *Recall*, *Precision*, *Balance-rate* and the confusion matrix. The 1σ uncertainty of each value is estimated by adding/subtracting the *mean-prediction-error* from the averaged predictions, and then casting again.
5) When applying the network on real CANDELS images, we use the same *mean-prediction-error* from the testing-set on the CANDELS prediction-vector, and estimate the uncertainty of the results in the same way.

Explanation: The accuracy is the percentage of correct predictions, which means that this is the probability of a random sampled prediction to be correct. It defines a Binomial distribution over the predictions: If we sample *n* predictions, and repeat the process *k* times, then the mean and 1-standard deviation of the number of correct predictions (from *n* samples) is given by:

$$mean = n\cdot p = n\cdot\overline{acc} \qquad (14)$$

$$\sigma = \sqrt{np(1-p)} = \sqrt{\left(\mathrm{n}\cdot\overline{acc}\cdot(1-\overline{acc})\right)} \qquad (15)$$

σ is the standard deviation of the total number of correct predictions. In other words, it is the uncertainty of the mean number of correct predictions if we repeat the *n* sampling *k* times. By dividing the 2 values by *n* we get the mean-probability and mean-standard deviation of a random prediction to be correct, over *k* runs:

$$\text{mean-probability} = \frac{mean}{n} = \overline{acc} \qquad (16)$$

$$\text{mean-prediction-error} = \frac{\sigma}{n} = \sqrt{\overline{acc}\cdot\frac{1-\overline{acc}}{n}} \qquad (17)$$

Note:

We can see in the results section 4 that the error estimation we get is usually very small. It means that the performance of our networks together with our technique of averaging over different backgrounds successfully give results with a high accuracy, a high confidence and a small uncertainty, such that images barely switch classifications after adding/subtracting the prediction-error (meaning they rarely cross the 50% probability to either side). However, the scatter in the merger-fraction and merger-rate graphs are too big for the error-bars, which means that indeed we don't account for the whole error. It could be either the epistemic uncertainty in the model, or a general difference between the simulated images and even the simulation itself, and the real CANDELS images.

4. Results

In this section we show the results we got from the neural network, and discuss their interpretations. The plan is as following:

1. (4.1): We start with reproducing the results from Ferreira et al. 2020, who used a different simulation and data (IllustrisTNG-300 and different CANDELS fields), which serves as a sanity check for our techniques. That includes the image sampling, the adding of backgrounds, the network structure and the HORIZON-AGN simulation as a whole. The results of Ferreira et al. (2020) in turn are claimed to match with Duncan et al. (2019) and Mundy et al. (2017), who used catalogs that were labeled manually from the SDSS survey. By reproducing similar results with only a fraction of the data that Ferreira et al. 2020 used, and without human bias in the labeling that might happened in Duncan et al. (2019) and Mundy et al. (2017), we show the supremacy of the Horizon-AGN simulation that compensates for its smaller size by having better spatial and temporal resolutions.
2. (4.2): Find the time-scale for the entire major-merger event. First we show in subsection 4.2.1 an empirical justification from the simulation for working with dynamical times instead of absolute times (in addition to the theoretical justification given in the introduction (1.2)). Then, in subsection 4.2.2 we verify that our network performs also with a small data in a narrow redshift range of $0.5 \leq z \leq 1$. Then, in subsection 4.2.3 we find from the simulation the best measure for the end time of the Post-Merger phase (until relaxation) in dynamical time units. This is a completely objective and unbiased measure of the timescales, which is missing in previous works.
3. (4.3): From the simulation identify the middle phase of the merger (the "In-Merger"), from first passage to coalescence, and evaluate its duration. As far as we know, we are the first to do it.
4. (4.34.4): Find the new merger-fraction and merger-rate in real observations (CANDELS), while retaining the ability to distinguish between Pre-Mergers and (In-Merger + Post-Mergers). All with dynamical time units. As far as we know, this is the first time that it is done using Deep-Learning and dynamical-timescales, which makes it the most accurate measure done so far.

In each case we first train our network on a training-set and test it on a separate testing-set, both from the simulation, where we know the true labels (the ground truth). We measure the success of the network over the testing-set, using the metric explained in section 3.2. The data-sets are balanced by class/phase and in most of the cases also by redshift bins, meaning that we construct them by randomly sampling an equal or nearly equal number of images from each class and each redshift bin. It reduces the number of images we work with, but eliminates a systematic bias towards any class/redshift that the network might learn because of originally non-uniform distributions.

After the training we evaluate the model over a testing set, resulting in a prediction/probability vector and an 1σ error estimation for the predictions-error according to the recipe that was described in section 3.3 (high accuracy and confidence lead to a low prediction error). In order to calculate values such as the Precision, Recall and the confusion matrix shown (examples: Table **4** and Figure **5**), we project each prediction to a binary result (for example: **merger** (> 0.5) or **no-merger** (< 0.5)) and compare it to the true label that we already know from the simulation. We calculate the uncertainty of these values by applying the prediction error from above to the predictions vector before the projecting, and only then project it to the binary labels, it may change some of the final labels which gives the error estimation for the metric values. Notice that applying the prediction error may or may not change the final label of each image, depending on the size of the error and the confidence of the prediction. For

example: If a galaxy is predicted to be a **merger** with 0.89 probability, and the prediction-error is 0.05, it will still remain a **merger** and the Precision and Recall remain the same.

In the same way, when applying the network model to real CANDELS images, we get a vector for their predictions as well. The errors are estimated similarly by adding/subtracting the same 1σ prediction-errors that we got from the evaluation over the simulated testing-set, to/from the predictions of the real CANDELS, and then projecting each prediction to either class.

Remember that this error is just the uncertainty over the data, which results from variations over the image sampling, noise and backgrounds. It does not include uncertainty over the quality of the network model, its architecture and learning.

General parameters and filters:

- Galaxies with stellar mass: $M_* \geq 10^{10} M_\odot$.
- Mergers mass ratio: $\mu \geq \frac{1}{4}$.
- Redshift range: $0.5 \leq z \leq 2$.
- Balancing by 3 redshift bins: $[0.5, 1, 1.5, 2]$.
- HST filters: F105, F160, F435, F606, F850.
- The distance between 2 Major-Mergers progenitors is less than $20\ kpc$, to make sure we capture both of them within the image frame.

4.1. Comparing to a previous work with another simulation

In this part we wish to reproduce the results obtained in Ferreira et al. 2020. Similar to them, we used a machine learning technique in 2 layers to classify Major-Mergers (**merger**) vs No-Mergers (**no-merger**), and then Pre-Mergers (**preM**) vs the combined In and Post Mergers (**inM + postM**). While Ferreira et al. 2020 trained and tested it on IllustrisTNG300 cosmological simulation, we did it on Horizon-AGN.

Differences between the simulations:

The IllustrisTNG300 simulation has 2×2500^3 spatial resolution elements over a volume of $(300\ h^{-1}Mpc)^3$, down to 1.4 proper kpc and a time resolution of $\sim 10^8$ years. Ferreira et al. 2020 had a total number of $\sim 62{,}000$ images and they used the filters **F125** (orange) and **F160** (red).

The Horizon-AGN simulation has 2×1024^3 spatial resolution elements over a volume of $(100\ h^{-1}Mpc)^3$, down to 1 proper kpc, time resolution of $\sim 1.7 \cdot 10^7$ years (so both the spatial and temporal resolutions are higher). We have a total number of $\sim 13{,}000$ images (depending on the exact case that we run) and we use the filters **F105** (yellow), **F160** (red), **F435** (purple), **F606** (blue) and **F850** (green).

We use much less images but with a higher resolution and more band filters. Other than that, we reproduced their results by using similar parameters which include the parameters listed at the top of this page and the same observational time window for each case.

4.1.1. Mergers vs No-Mergers – Comparing the merger-fraction and merger-rate to previous work

We train a classifier for Mergers vs No-Mergers with a fixed observation time-window in absolute time units, and produce the merger-fraction and merger-rate as a function of redshift.

Simulation parameters and results:

We trained and tested the network on the simulation's mock images with the following parameters:

- Mergers [-0.3 – 0.3] vs no-mergers [0.5+] in Gys units.
- Training-set size: 11,088 images.
- Testing-set size: 2,160 images.
- Among the **no-merger** images, 17% (918/5,544) in the training-set and 13% (140/1,080) in the testing-set, are "late" images from the Major-Mergers catalog.

Redshift	Number of images	phase	Number of images
$0.4 \leq z < 1$	4,201	Pre-Merger	2,772
$1 \leq z < 1.5$	3,532	inM + postM	2,772
$1.5 \leq z < 2$	3,355	No-Merger	5,544

Table 3: *The distribution of the training-set's mock images from the simulation after they were balanced by redshift bins (**left**) and phase (**right**). The case is of **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] in Gys units (not dynamical time). Notice that the number of **no-merger** images equals to the number of **preM + inM + postM**. However, there are slightly more images in lower redshifts than in higher redshifts (up to 20% difference at maximum). The high score means that this imbalance isn't impactful.*

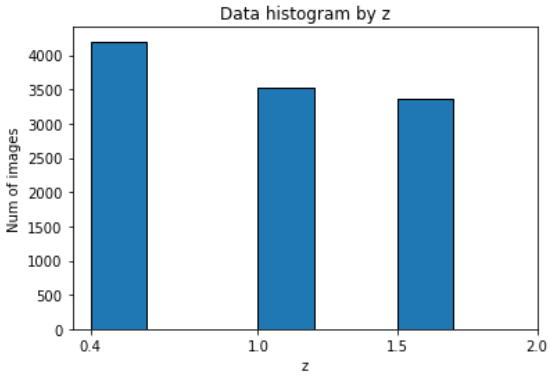


Figure 4: *A histogram the shows the distribution of the training-set's mock images from the simulation by redshift bins, after balancing. The case is of **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] in Gys units.*

Table **3** and Figure **4** show the number of images in the training-set, and their distribution over redshifts and phases. We can see that they are perfectly balanced with respect to phase (number of **merger** = **no-merger** and the numbers of **preM** and (**inM + postM)** are equal), but with respect to redshifts there's a difference of up to 20% between the largest bin and the smallest one. The reason for this is that some of the **no-merger** images, specifically the "late" snapshots ($t > 0.5\ Gys$) that were drawn from the "Major-Mergers" catalog (see section 2.2.1), are a very small group so we had to keep as many of them as we could, while compromising a little on the balancing by redshift bins. The high score means that this imbalance isn't impactful.

Table **4** shows the results according to the 2 interchangeable metrics: "Accuracy" and "Balance/difference-rate", "Precision" and "Recall". We can see that all the results are above 90% with a very minimal error, which means the network learned very successfully. Figure **5** shows the confusion matrix, which visualizes the results in a clearer way. Each row in the confusion matrix represents the true labels of the relevant class, and each column represents the predicted label. The results are highly balanced with a small bias (0.009 ± 0.002) toward the **no-merger** class, and the small error is the result of the fact that the network is very confident in the predictions (see Figure **6**, so even after adding/subtracting the prediction error for each prediction (which is also small because the accuracy is very high), most of the galaxies don't switch classes ($> 50\% \Rightarrow$ **merger** ; $< 50\% \Rightarrow$ **no-merger**).

Figure **7** shows a sample of images classified by the network. From there we can see that **merger** galaxies are characterized by visible tidal features or other distorted morphology, like asymmetries between the shape of the core and the rest of the galaxy, or a pair of galaxies that both of them are bright in all the bands (and in many cases they also show distorted morphology). In contrast, **no-merger** galaxies look more "nice", symmetric and they tend to be fainter in the F606 (blue) and F435 (purple) bands than in the F160 (red) and F105 (yellow) bands, especially in the outskirts, as oppose to **merger** galaxies. These characterizations aren't always true though, which cause the confusions.

Metric	Score
Mean prediction error	0.0053
Accuracy	0.940 ± 0.005
Balance-rate	0.991 ± 0.002
Precision	0.937 ± 0.001
Recall	0.945 ± 0.002

Table 4: *The score of the **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] classification in Gys units, over the simulation's testing-set. The Metrics' meaning are explained in section 3.3 and the errors represent 1σ prediction-error estimation added to / subtracted from the predictions.*

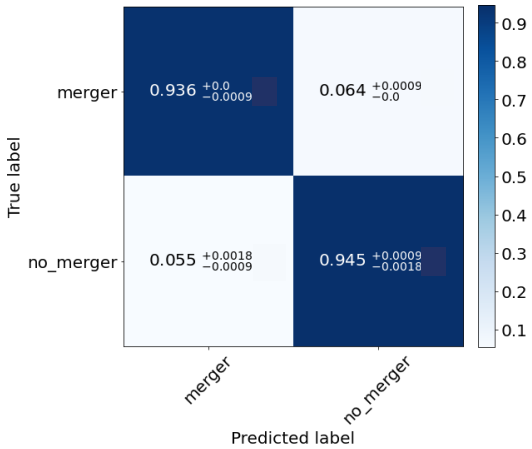


Figure 5: *Normalized confusion matrix for the **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] classification results in Gys units, over the simulation's testing-set. Each row represents the true labels of the relevant class, and each column represents the predicted label. The results are highly balanced with a small bias (0.009±0.002) toward the **no-merger** class. The small errors mean a high confidence for the predictions.*

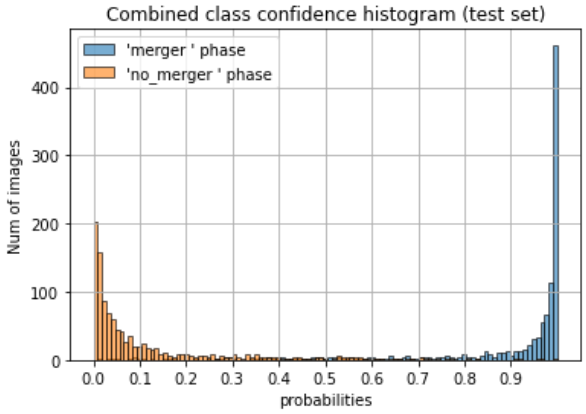


Figure 6: *Probability distribution of the prediction results of the **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] classification in Gys units, over the simulation's testing-set. The high confidence means that almost all the galaxies retain their classification even after accounting for the prediction error, leading to a small error in the confusion matrix. The classifier is somewhat more confident in the **merger** class than in the **no-merger** one, although according to the confusion matrix the accuracy of the **no-merger** class is higher.*

Summary:

*We trained a model to classify **mergers** [-0.3 – 0.3] vs **no-mergers** [0.5+] in absolute Gys units and tested it with very high accuracy (0.940 ± 0.005), high balance (0.991 ± 0.002) and high confidence. We measured the error range by applying the prediction-error to every prediction result before projecting it again to one of the classes, and almost none of them changed classes. **merger** galaxies have distorted morphology, or a pair of bright galaxies in all the bands (in many cases they're also distorted). **no-merger** galaxies look "nice", symmetric and fainter in the bluer bands than in redder ones, especially in the outskirts.*

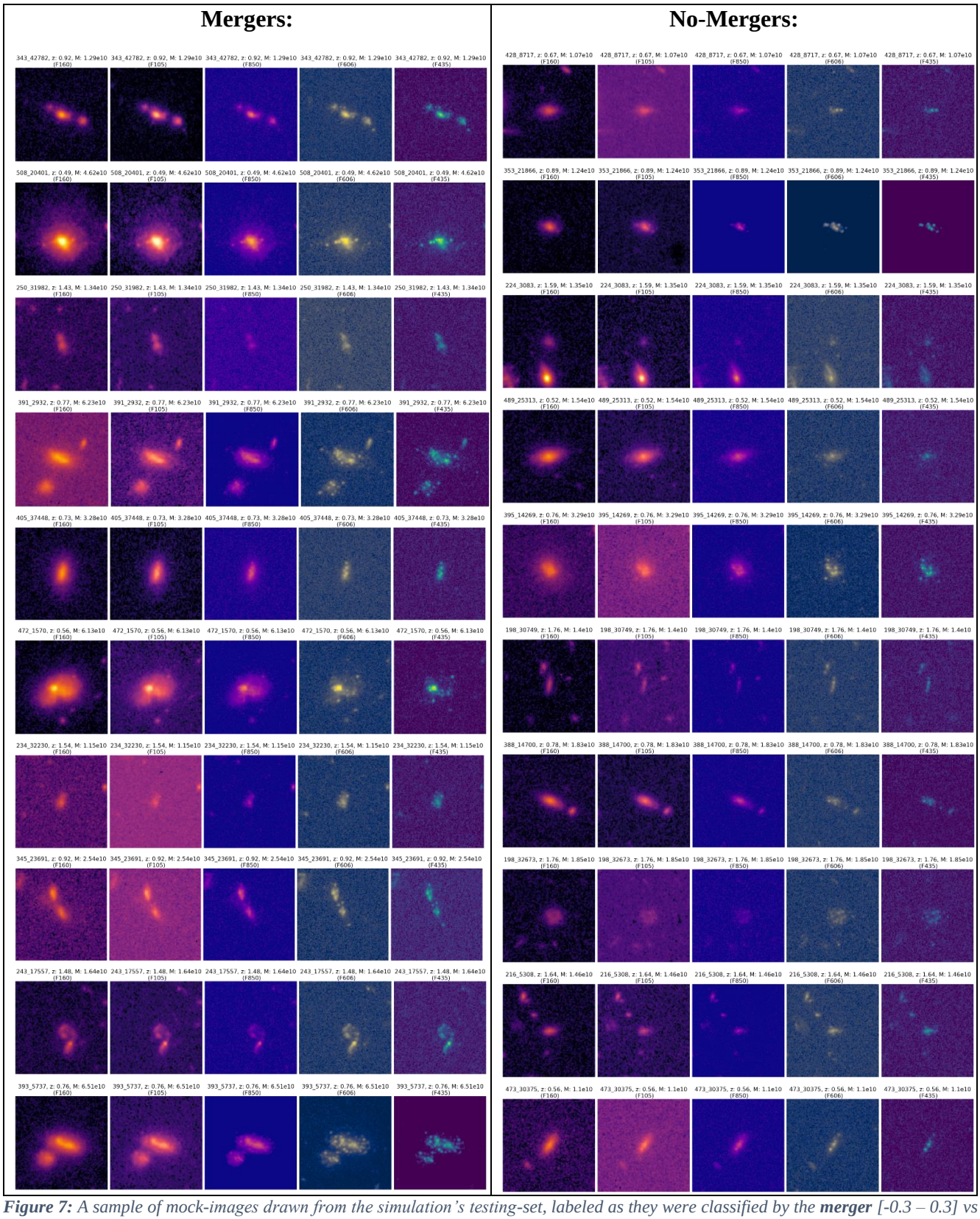


Figure 7: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.5+] classifier in Gys units. Mergers on the left side, no-mergers on the right side. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale.*

Real CANDELS results:

We used the trained **merger** vs **no-merger** classifier model on real CANDELS galaxies, and compared it to Ferreira et al. 2020 work. We used 573 images from 2 CANDELS fields: GOODS-S and GOODS-N. Ferreira et al. 2020 used 3,759 images from 5 CANDELS fields.

Table **5** shows the number of real CANDELS galaxies and their percentage that were classified in each class (Mergers and No-Merger). Again, high confident predictions with a small error estimation leads to a small number of galaxies that switch classes (1/73 in our case).

Phase	Total amount	Percentage
Mergers	73^{+0}_{-1}	$0.127^{+0}_{-0.002}$
No-Mergers	500^{+1}_{-0}	$0.873^{+0.002}_{-0}$

Table 5: *The total number of real CANDELS galaxies and their percentage that were classified for each phase by the* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.5+] classifier in Gys units. The errors are estimated by adding/subtracting the 1σ prediction-errors from the simulation, to/from the prediction results of the real CANDELS, and then projecting each prediction to either "Merger" (> 0.5) or "No-Merger" (< 0.5) phase. High confident predictions with small errors lead to barely any change in the final phase.*

We sorted all the galaxies into several redshift bins and calculated the merger-fraction and merger-rate within each redshift, as well as the uncertainty that comes from the galaxies that switched labels.

In Figure **8** we plot our merger-fraction as a function of redshift (red dots with the uncertainty as the error-bars) and add the best fit (orange line) with 1σ error of the fit (orange shaded area), in a half logarithmic scale (left) and a full logarithmic scale (right). For comparison, in the same graphs we also plot Ferreira's merger fraction (green stars), best fit (purple line) and 1σ error (purple shaded area). The formula of our best merger-fraction fit is given on top the panels.

In Figure **9** we plot our merger-rate as a function of redshift, by dividing the merger fraction with the observation time (0.6 Gys). We add the best fit with 1σ error of the fit, in a half logarithmic scale (left) and a full logarithmic scale (right). The color coding is the same as for the merger fraction in Figure **8**. The formula of our best merger-rate fit is given on top the panels. The merger-rate in equation (19) does not fully match the theoretical estimation in equation (5), but the power-law is very close (we got minimum 2.7 within 1σ and equation (5) estimates 2.5). We didn't expect a full match because equation (5) estimates the full halo accretion rate and we only look at galaxy major-mergers of a limited mass-cut ($M_* \geq 10^{10} M_\odot$). Within 2σ range we do match. Our merger-rate's magnitude at $z = 2$ also matches the theoretical merger-rate from equation (6) at the same redshift (0.45 Gyr^{-1}).

We can see in all the plots that our results and the results of Ferreira et al. 2020 (data and fits) are in a good agreement, as they both fall within each other's error range. And the results of Ferreira et al. 2020 in turn are claimed to match with Duncan et al. (2019) and Mundy et al. (2017). However, the scatters of our data are larger than of Ferreira et al. 2020, which could be the result of us having 6.6 times less images as them, and the error bars account for variations in the images only, not the uncertainty in the model, which is probably why they don't scale up to the scatter.

$$MF(z) = (0.009 \pm 0.005) \cdot (1+z)^{3.26 \pm 0.38} \quad (18)$$

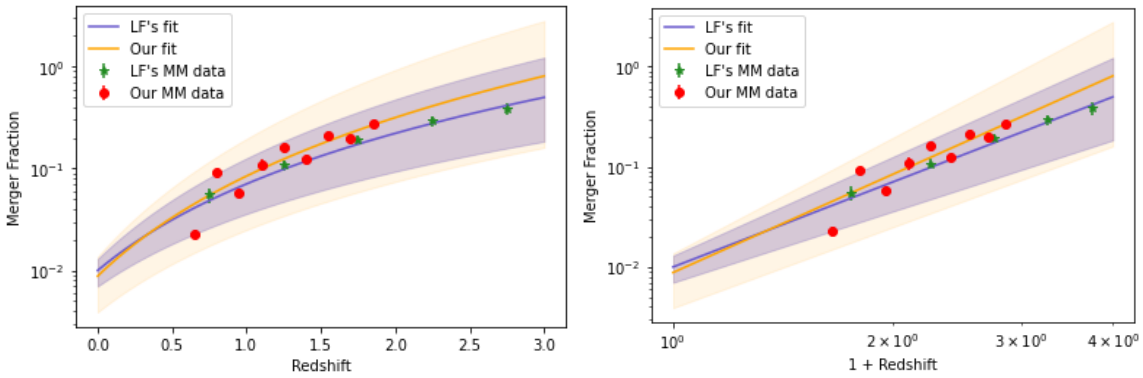


Figure 8: *Left: Merger-Fraction vs redshift in a half-log scale. Right: Merger-Fraction vs (1+redshift) in a log-log scale. The case is of* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.5+] in Gys units, over real CANDELS images. The lines are the best fit and the shaded areas are the 1-σ error-range of the fits. We add the results from Ferreira's paper (green and purple) for comparison and we can see that there's a good agreement.*

$$MR(z) = (0.015 \pm 0.008) \cdot (1+z)^{3.26 \pm 0.58} [Gyr^{-1}] \quad (19)$$

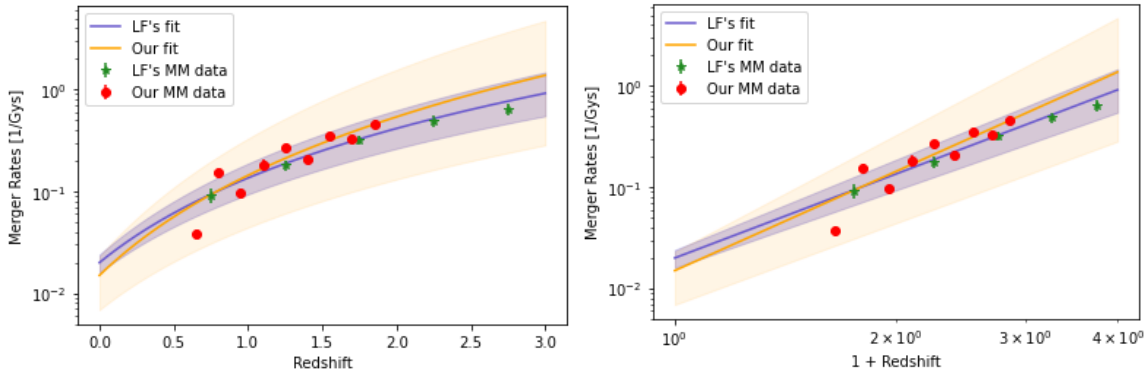


Figure 9: *Left: Merger-Rate vs redshift in a half-log scale. Right: Merger-Rate vs (1+redshift) in a log-log scale. The case is of* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.5+] in Gys units, over real CANDELS images. The lines are the best fit and the shaded areas are the 1-σ error-range of the fits. We add the results from Ferreira's paper (green and purple) for comparison and we can see that there's a good agreement.*

Summary:

We applied the merger vs no-merger classifier (absolute times units) on real CANDELS images, resulting in 73^{+0}_{-1} */ 573 mergers. We sorted them into redshift bins and calculated the merger-fraction and merger-rate within each bin. The error range is estimated by applying the prediction-error to each image before projecting it again to one of the classes. The merger-fraction and merger-rate are in good agreement with Ferreira et al. 2020, but with a bigger scatter, probably due to smaller data. Our merger-rate's magnitude also matches the theory in equation* (6) *at* $z \sim 2$ *(*$0.45\ Gyr^{-1}$*) and the power law (minimum 2.7) is not much larger than the theoretical value in equation* (5) *(2.5), which was expected.*

4.1.2. Pre-Mergers vs combined (In-Mergers + Post-Mergers)

We train a classifier for Pre-Mergers vs the combined (In-Mergers + Post-Mergers) phases with a fixed observation time-window, for the galaxies that were previously classified as mergers.

Simulation parameters and results:
We trained and tested the network on the simulation's mock images with the following parameters:

- Pre-Mergers [-0.3 – 0] vs (In-Mergers + Post-Mergers) [0 – 0.3] in Gys units.
- Training-set size: 7,056 images.
- Testing-set size: 756 images.

Redshift	**Number of images**	**phase**	**Number of images**
$0.4 \leq z < 1$	2,352	Pre-Merger	3,528
$1 \leq z < 1.5$	2,352	In+Post-Merger	3,528
$1.5 \leq z < 2$	2,352		

Table 6: *The distribution of the training-set's mock images from the simulation after they were balanced by redshift bins (**left**) and phase (**right**). The case is of **preM** [-0.3 – 0] vs **(inM + postM)** [0 – 0.3] in Gys units.*

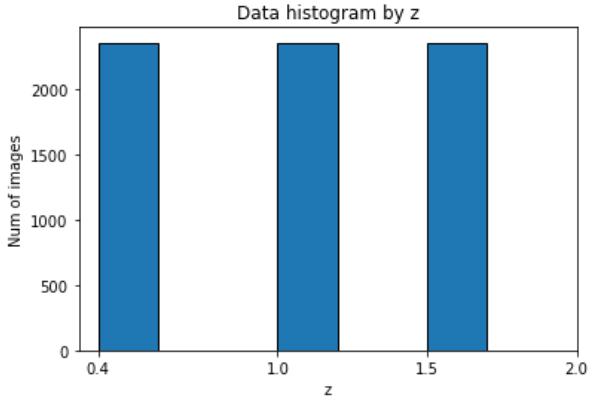


Figure 10: *A histogram the shows the distribution of the training-set's mock images from the simulation by redshift bins, after balancing. The case is of **preM** [-0.3 – 0] vs **(inM + postM)** [0 – 0.3] in Gys units.*

Table **6** and Figure **10** show the number of images in the training-set, and their distribution over redshifts and phases. We can see that they are perfectly balanced with respect to phase (**preM** = **inM** + **postM**) and redshift (bin1 = bin2 = bin3).

Table **7** shows the results according to the 2 interchangeable metrics: "Accuracy" and "Balance/difference-rate", "Precision" and "Recall". We can see that the performance is less good and less balanced than in the **merger** vs **no-merger** case, indicating that classifying between inner phases of a merger is a harder task (also the data-sets are smaller which makes it harder), but still good enough to work with. Figure **11** shows the confusion matrix, which visualizes the results in a clearer way. The results have a confusion in average of 0.3 between the classes, and are somewhat balanced with a bias (of 0.11 ± 0.026) toward the (**inM** + **postM**) class. The error is of the scale of 5%-10% of the value, which is the result of the network being not very confident in the predictions (see Figure **12**).

Figure **13** shows a sample of images classified by the network. From there we can see that roughly, the network sorts out the relevant galaxies from the backgrounds by taking only galaxies which are more dominant and equally bright in all the bands. Then if it has 1 galaxy it's classified as a (**inM** + **postM**)

and if it has 2 it's a **preM**. Confusion happens mostly when a **preM** image shows 1 galaxy (maybe the other one hides behind the first one?), or when a (**inM** + **postM**) image shows several galaxies (usually within a short time after the merging. Probably the **inM** phase) or when there are many objects that are equally bright and appear in all the bands. These characterizations aren't always true though. Figure **14** shows a sample of incorrect classifications.

Metric	Score
Mean prediction error	0.017
Accuracy	0.707 ± 0.017
Balance-rate	0.891 ± 0.026
Precision	0.687 ± 0.006
Recall	0.762 ± 0.013

Table 7: *The score of the **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in Gys units, over the simulation's testing-set. The Metrics' meaning are explained in section 3.3 and the errors represent 1σ prediction-error estimation added to / subtracted from the predictions.*

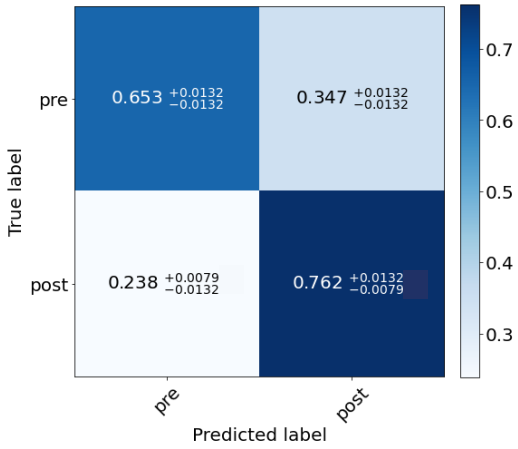


Figure 11: *Normalized confusion matrix for the **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in Gys units, over the simulation's testing-set. Each row represents the true labels of the relevant class, and each column represents the predicted label. We can see a confusion between the classes with a bias of 0.109 toward the (**inM** + **postM**) phase, indicating that the 2 phases sometimes look alike and the network straggles more with the **preM** phase.*

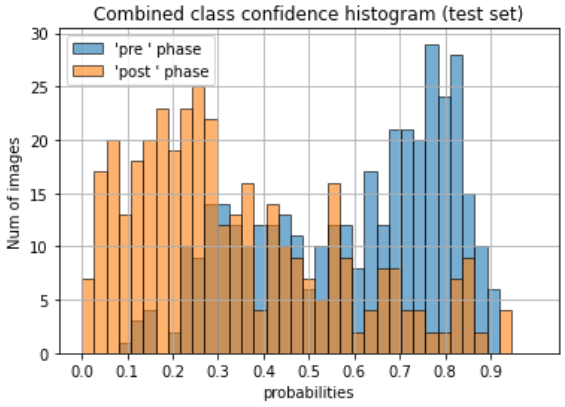


Figure 12: *Probability distribution of the prediction results of the **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in Gys units, over the simulation's testing-set. There are not a few cases of confusion, indicating that the 2 phases sometimes look alike. Could be around the merging at $t = 0$.*

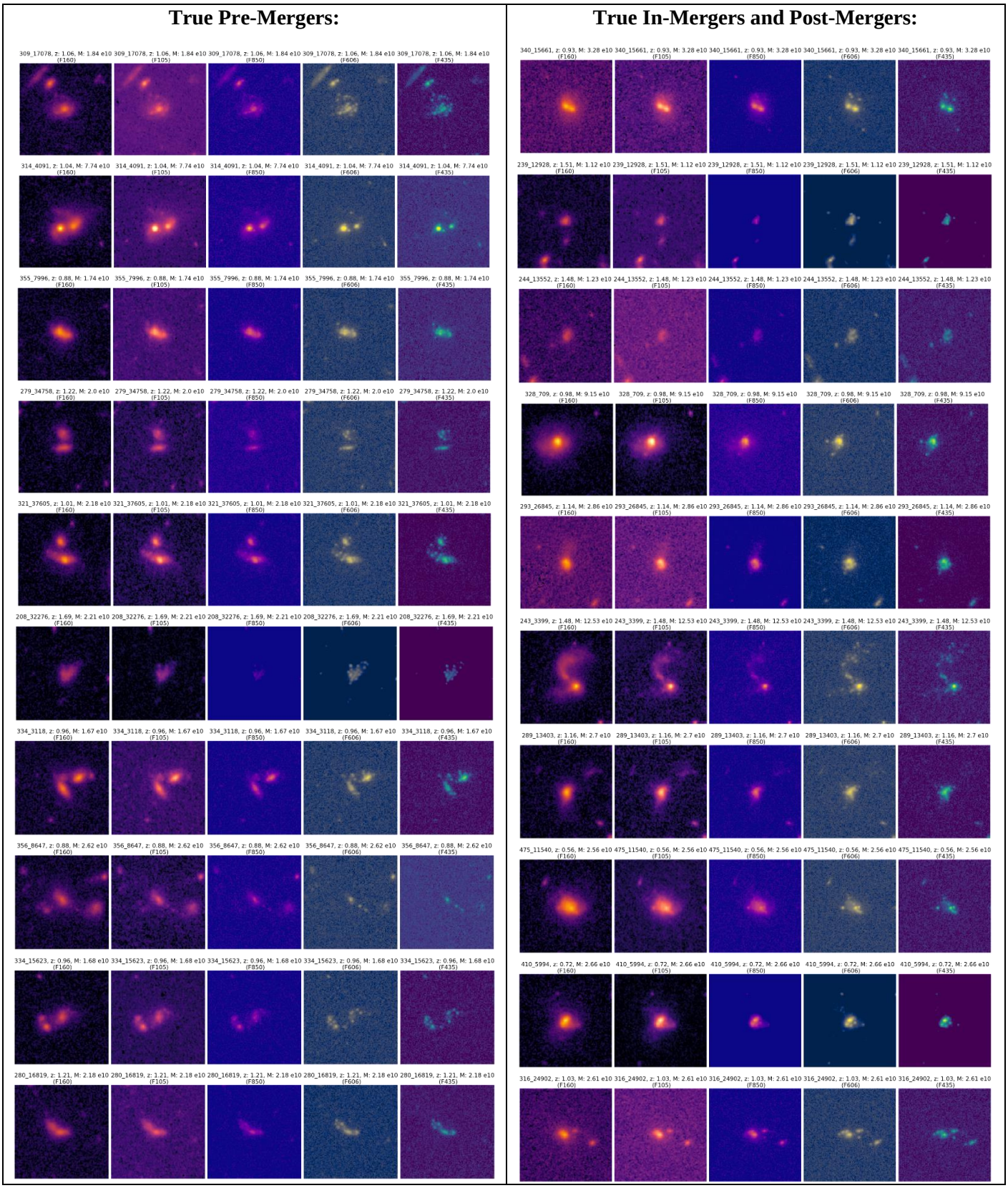


Figure 13: *A sample of mock-images drawn from the simulation's testing-set,* ***correctly*** *classified by the* ***preM*** *[-0.3 – 0] vs* ***(inM + postM)*** *[0 – 0.3] classifier in Gys units. Pre-Mergers on the left side, (In-Mergers + Post-Mergers) on the right side. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale.*

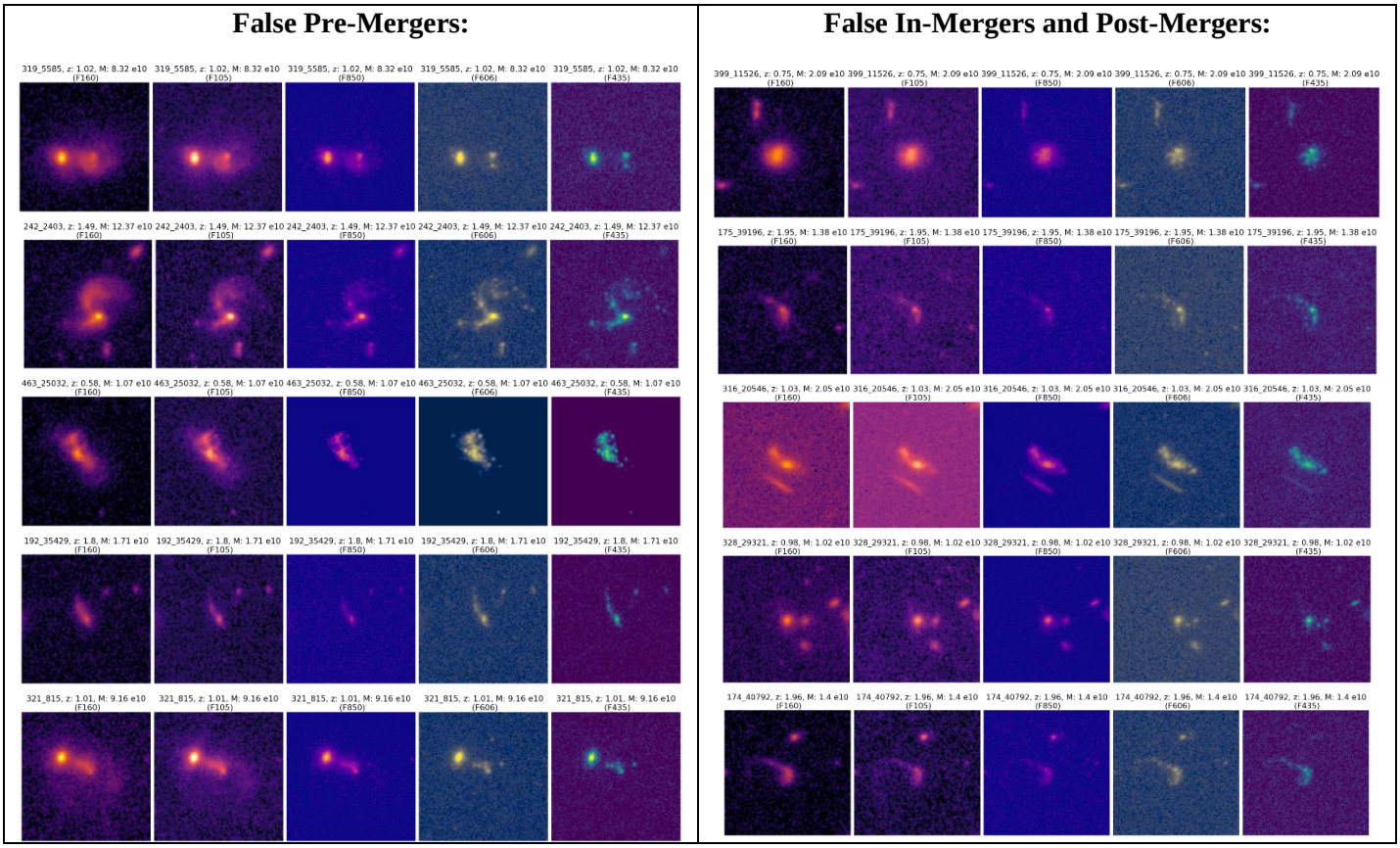


Figure 14: *A sample of mock-images drawn from the simulation's testing-set,* ***wrongly*** *classified by the* ***preM*** *[-0.3 – 0] vs* ***(inM + postM)*** *[0 – 0.3] classifier in Gys units. False Pre-Mergers on the left side, false [In-Mergers + Post-Mergers] on the right side. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale.*

Real CANDELS results:

We then used the trained **preM** vs (**inM** + **postM**) classifier model on real CANDELS galaxies, and compared it to the results in Ferreira et al. 2020. We took only the galaxies that were previously classified as Mergers by the **merger** vs **no-merger** classifier (Gys units), classified them into **preM**s or (**inM** + **postM**)s and compared the ratio of our **preM** / (**inM** + **postM**) within different redshifts and in total, to what Ferreira et al. 2020 got. Since we defined the duration time of each phase to be symmetric (0.3 Gys), then we expect a ratio of $preM/(inM + postM) \rightarrow 1$

Table **8** shows the number of real CANDELS **merger** galaxies and their percentage that were classified in each class (**preM** and **inM** + **postM**), with 2σ prediction-errors that leads to 8/73 galaxies that change their labels. We see a very balanced result with a ratio of just $preM/(inM + postM)\,\big|_{our} = 1.09^{+0.34}_{-0.17}$, which is better on average than what Ferreira et al. 2020 (our error range is of 2σ and does include the results from Ferreira et al. 2020): 1.32 for $0.5 \leq z \leq 3$; and 1.42 for $0.5 \leq z \leq 2$.

Phase	Total amount	Percentage
Pre-Mergers	38^{+5}_{-3}	$0.521^{+0.068}_{-0.041}$
In+Post-Mergers	35^{+3}_{-5}	$0.479^{+0.041}_{-0.068}$

Table 8: *The total number of real CANDELS galaxies and their percentage that were classified for each phase by the* ***preM*** *[-0.3 – 0] vs* ***(inM + postM)*** *[0 – 0.3] classifier in Gys units. We only considered galaxies that were classified as "Mergers" by {4.1.1}. The errors are estimated by adding/subtracting the 2σ prediction-errors from the simulation, to/from the prediction results of the real CANDELS, and then casting each prediction to either Pre-Merger (> 0.5) or (In-Mergers + Post-Merger) (< 0.5) phase. High confident predictions with small errors lead to barely any change in the final phase. Notice that there's a small bias toward the* ***preM*** *class, although in the simulation we got a small bias toward the* ***(inM + postM)*** *class. It means that the small sample of 73 galaxies isn't as representative as the 756 galaxies in the testing-set.*

We sorted the galaxies into several redshift bins. In Figure **15** we plot the fraction of galaxies that were classified as **preM** (blue down-rectangles) and the fraction of galaxies that were classified as (**inM** + **postM**) (orange up-rectangles), within each redshift bin, normalized such that their sum is always 1. The left panel is of our results and the right panel is of the results from Ferreira et al. 2020, for comparison. The error bars in the left panel (ours) are the result of the 2σ uncertainty in the prediction, as described above. Ferreira's error bars were calculated in a different way and represent 1σ uncertainty in Monte-Carlo error estimation (read their paper to find out more).

In Figure **15** left panel we can see that in exactly half of the cases the majority of the galaxies are **preM** and in the other half they are (**inM** + **postM**), and it seems randomly ordered. In addition, in 5/6 of the cases the gap between **preM** and (**inM** + **postM**) is smaller than 0.3, and in one case ($1 < z < 1.2$) it is even 0. In one case though the gap is as big as 0.6 ($0.8 < z < 1$). The gaps in the left panel (our) seem to be a bit larger than in the right panel (of Ferreira et al. 2020), but of the same scale. Given the fact that in our work (left panel) we are talking about just 12 galaxies on average for each bin (so ~6 for **preM** and ~6 for (**inM** + **postM**)) we think that it is a satisfying result that shows that our network classifies successfully **preM** vs (**inM** + **postM**). However, one might claim that these results also support a random guessing, but the results from the simulation tell us that the network **does not** try a random guessing strategy.

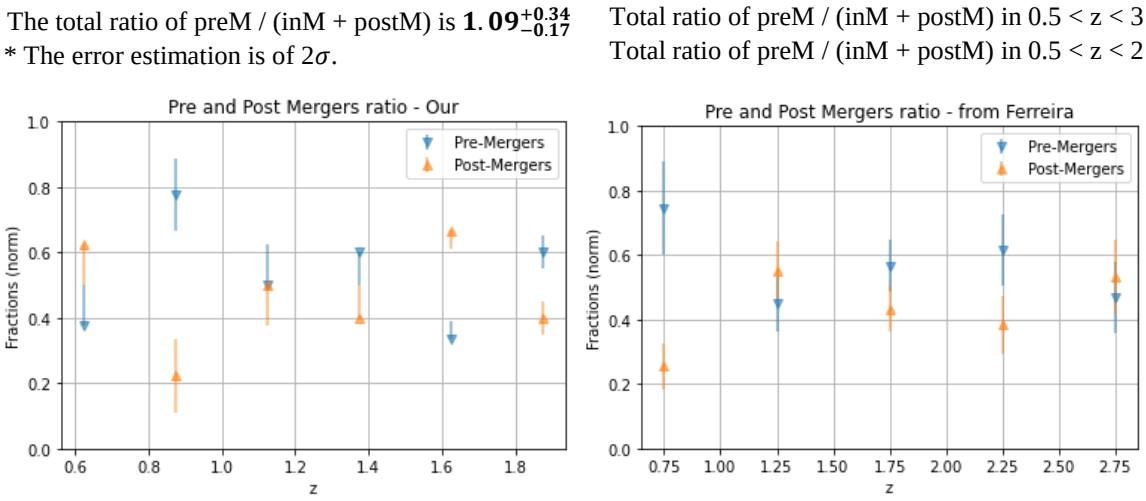


Figure 15: *The normalized fraction of **preM** and (**inM** + **postM**) predictions in our work (left) and Ferreira's (right), among the real CANDELS galaxies that were classified as "Mergers" by {4.1.1}, in different redshift bins. The case is of **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in Gys units. The errors in our work represent 2σ over the predictions, which are then projected to either **preM** (> 0.5) or (**inM** + **postM**) (< 0.5). The facts that our results are close in most of the bins, that they flip from **preM** dominance to (**inM** + **postM**) dominance randomly and that the ratio Pre/Post is closer to 1 than in Ferreira's results, are indications of a well-balanced classification.*

Since there's a good agreement between our merger-fraction and merger-rate and those from Ferriera et al. 2020, and we were able to distinguish between Pre-Mergers and (In-Mergers + Post-Mergers) just as well and even better, when using similar definitions and parameters, it strengthens our confidence in the relevance of both cosmological simulations - IllustrisTNG and Horizon-AGN, as well as in our network and our data and backgrounds sampling techniques. With this in mind we can start exploring different models that use observation time window that depends on the halo's dynamical time and scales with redshift.

Summary:
*We trained a model to classify **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] in absolute Gys units and tested it with accuracy of 0.707 ± 0.017, balance 0.891 ± 0.026 and moderate confidence. The lower performance lead to a higher error-range than in the previous case. First the network sorts out galaxies that are distorted or bright in all the bands, then if there's only 1 it's usually (**inM** + **postM**) and if more it's **preM**. Then we applied the model to real CANDELS galaxies that were classified as mergers. There isn't a visible bias in different redshifts and* 38^{+5}_{-3} / 73 *were classified as **preM**, which is a better ratio* ($1.09^{+0.34}_{-0.17}$ with 2σ error estimation) *than in Ferreira et al. 2020 (1.32 for* $z \leq 2$*).*

4.2. Measuring the Major-Merger Duration

In section 4.1 we studied the merger events, the merger fraction and rate and separated the Pre-Mergers from the rest of the merger phases, by using a fixed duration time window of 0.6 Gys (0.3 Gys for **preM** and 0.3 for (**inM** + **postM**)). However, as discussed in the introduction (1.2), the time that takes to a major merger event to complete is expected to vary through redshift, so by using a fixed time we might be mixing galaxies of different phases. We therefore wish to use a different, redshift depended definition for the observation time windows that changes as the halo's dynamical time changes (because this is the time scale that on average should govern gravitational based processes at different redshifts). It means that we can work with a unitless parameter t/t_{dyn} that takes the same value for all redshifts, instead of using absolute time (Gys). In the next parts we will show you an empirical justification for using the theoretical equation *(2)* for dynamical time in the simulation, and find the best definition for the duration of the major-merger event, in dynamical time units.

4.2.1. Dynamical Time dependence

In this part we wish to show a proof of concept for the dynamical time dependence of the merger event within the Horizon-AGN cosmological simulation. Equation *(2)* describes the dark-matter halo's dynamical time as a function of redshift. Meaning the typical free fall time within a typical DM halo of a given redshift. But although we expect this theoretical expression to govern merger events in the real universe, it's worth checking, empirically, if this is also the case in the Horizon-AGN simulation. If not, it means that the sub grid physics of the simulation doesn't match the real universe.

We divided all the 273 Major-Merger events we have into redshift bins based on the redshift of the central snapshot ($t = 0$). For each z-bin we measured the mean and 1-STD time it takes for the 2 progenitors to fall from a distance of 4 × [the radius of the main progenitor] to the first passage, marking the beginning of the "In-Merger" phase. We plot the results as a function of the redshift in Figure **16** where the mean times of the redshifts are the blue stars and the 1-STD is the shaded area. The left panel is in absolute time units (Gys) and the right panel is in dynamical times units (all the times were normalized by the halo's dynamical time of the relevant redshift, according to equation *(2)*. In both panels we also add a rough fit as a constant fraction of the theoretical DM halo's dynamical time: $0.27t_{dyn}(z)$ (the orange line). Note: It is not a "best fit", just a rough one. It is enough for us to see that indeed the system evolves according to the theoretical dynamical time. Therefore, the duration of the merger's phases, and the merger event as a whole, should be defined in dynamical times units instead of absolute time units.

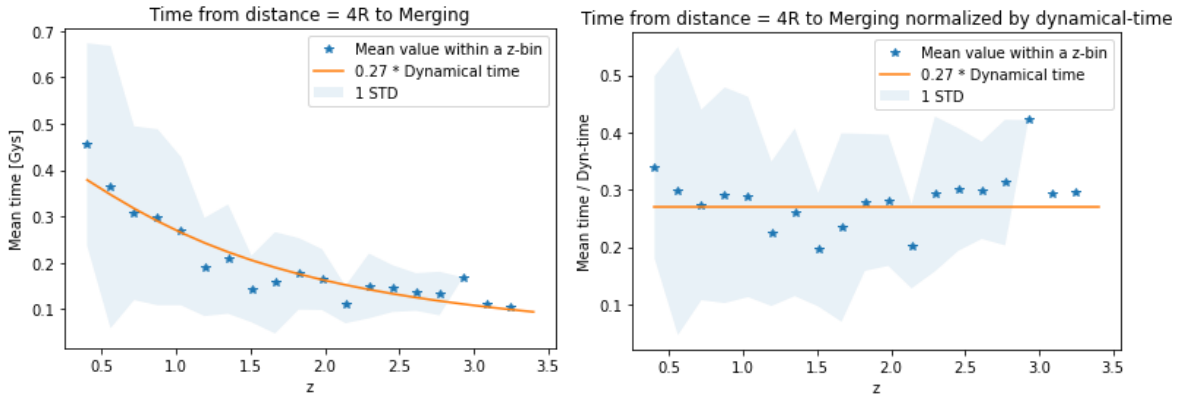


Figure 16: *The time it takes to pre-merging galaxies to get from a distance of* $(4 \times main\ progenitor's\ radius)$ *to the closest point of the first passage (the beginning of the "In-Merger" phase), vs redshift. Left: Absolute time units (Gys). Right: Normalized by the DM halo's dynamical time (equation (2)). Each point is the mean value of the galaxies within a small redshift bin, taken from the simulation, and the shaded area is 1 STD. The orange line represents* $(0.27 \cdot Dynamical\ times)$ *and is a rough fit that shows that the averaged time it takes to a merger to get from one stage (4R distance) to another stage (0 distance) scales with the DM halo's dynamical time.*

Summary:

We checked if the Horizon-AGN simulation follows the theoretical estimation of the dynamical times per redshift from equation (2), by checking the mean and 1-STD time that it takes to pre-merger galaxies in a given redshift to fall from a distance of $4 \times$ *[the radius of the main progenitor] to the first pericenter* $(t = 0)$*. Turned out it takes roughly* $0.27\ t_{dyn}$*, which means we can work with dynamical times units instead of absolute times.*

4.2.2. Reproducing Merger-Fraction and Merger-Rate for z ≤ 1 using absolute time units

In this part we reproduce the merger-fraction and merger-rate using absolute time units, but on a very narrow range of redshifts: $0.5 \leq z \leq 1$. The motivation for this is to verify that our network can deal with this narrow redshift range and small data, because we will use this range in section 4.2.3 to find the end-time of the post-merger phase as a fraction of the halo's dynamical time.

The problem: When we take images of galaxies of redshift up to $z = 2$ we need to balance them according to redshift bins by down sampling an equal amount for each z-bin, to prevent biasing over redshifts. But as shown in Figure **16**, in higher redshifts there are much less "late" images than in lower redshifts, so we end up throwing away many. In section 4.2.3 we train and test a classifier between "early" and "late" snapshots of the "major-mergers" catalog (meaning snapshots of early and late times since *t=0*), with different time-cuts between them, to see what is the best time to end the merger event. This classification already uses small data because it doesn't include pre-mergers or the isolated images from the isolated catalog. If in addition we redshift-balance them, we end up with a too small dataset.

The solution: We look only at images of redshifts $0.5 \leq z \leq 1$ (the largest z-bin) so we don't need to balance them, and since we know that the duration of the merger in dynamical time units should be consistent in other redshifts, we will extrapolate the result up to redshift $z \leq 2$.

But first, in this part we reproduce the merger-fraction and merger-rate up to redshift $z \leq 1$, **using absolute time units** (Gys), as in section 4.1, to verify that our network doesn't need the variety and can learn the relevant values for this redshift range, despite the smaller range and data.
Simulation parameters and results:

We trained and tested the network on the simulation's mock images with the following parameters:

- Mergers [-0.3 – 0.3] vs no-mergers [0.5+] in Gys units.
- $0.5 \leq z \leq 1$
- Training-set size: 9,648 images.
- Testing-set size: 624 images.
- Among the **no-merger** images, 16% (795/4,824) in the training-set and 31% (96/312) in the testing-set, are "late" images from the "major-mergers" catalog.

As in previous parts, the datasets are balanced by class and Table **9**, Figure **17** and Figure 18 show us the metrics' results, the confusion matrix and the probability confidence histogram, respectively, with the same color-coding. We see high scores with a very high confidence, but that the network over estimates the **no-merger** class over the **merger** class.

Metric	Score
Mean prediction error	0.0107
Accuracy	0.926 ± 0.011
Balance-rate	0.865 ± 0.019
Precision	0.876 ± 0.015
Recall	0.994 ± 0.000

Table 9: *The score of the **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] classification in Gys units, $z \leq 1$, over the simulation's testing-set. The Metrics' meaning are explained in section 3.3 and the errors represent 1σ prediction-error estimation added to / subtracted from the predictions. We can see that the results aren't very balanced.*

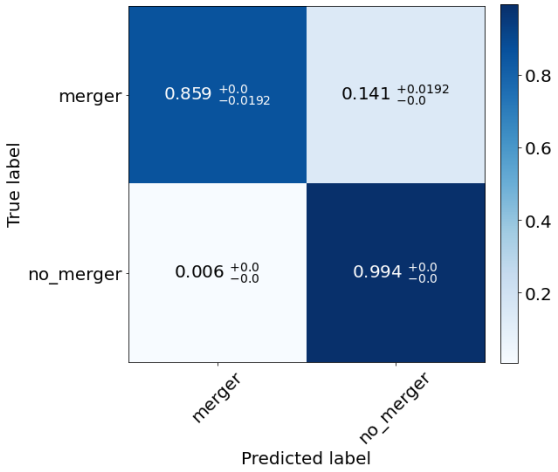


Figure 17: *Normalized confusion matrix for the **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] classification results in Gys units, $z \leq 1$, over the simulation's testing-set. Each row represents the true labels of the relevant class, and each column represents the predicted label. The results are less balanced than the case of classification till $z \leq 2$, with a bias (0.135) toward the **no-merger** class. The small errors mean a high confidence for the predictions.*

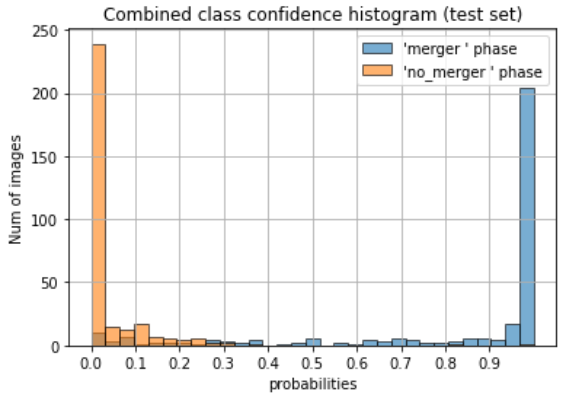


Figure 18: *Probability distribution of the prediction results of the **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] classification in Gys units, $z \leq 1$, over the simulation's testing-set. The classifier is very confident in both classes, which leads to a very small error in the score and confusion matrix.*

Real CANDELS results:

We then used the trained **merger** vs **no-merger** classifier model for redshifts $0.5 \leq z \leq 1$ on real CANDELS galaxies of the same range, and compared it to the results from Ferreira's et al. 2020. We got 25^{+0}_{-3} galaxies that were classified as mergers, we sorted them into narrower redshift bins and calculated the merger fraction and merger rate of each bin. In Figure **19** we plotted them (red dots) against the results from Ferreira et al. 2020 (green stars) and fit (purple line and purple shaded area). The Left panel shows the Merger-Fraction and the right panel shows the Merger-Rate. We can see that our results fall in both panels within the range of the results from Ferreira et al. 2020, which tells us that our network successfully learned the merger-fraction and merger-rate for this redshift in the given conditions, and therefore we can trust it to learn relevant "local" information even with a smaller dataset in a narrow redshift range.

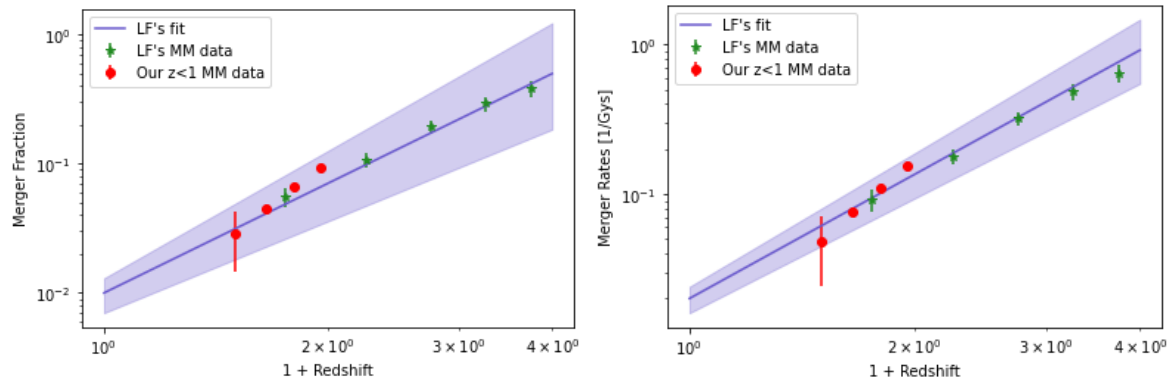


Figure 19: *Left: Merger-Fraction vs (1+redshift) in a log-log scale. Right: Merger-Rate vs (1+redshift) in a log-log scale. The 4 red points are our results of* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.5+] in Gys units, $0.5 \leq z \leq 1$, over real CANDELS images. For comparison, the results are plotted over Ferreira's results: data (green points), fit (purple line) and 1σ range (purple shaded area). One can see that even though we trained our network over a very narrow redshift range, it still recovered the right values for this range (they fall within the shaded area). Meaning we can trust the network in future exploration within the range of $z \leq 1$.*

4.2.3. The duration of the Major-Merger event

In this part we explore the best time cut for the end of the Post-Merger phases, and it will define the duration of the entire merger event (Pre-Merger will start at a symmetric timescale). We train several classifier models to distinguish "Early" snapshots from "Late" snapshots, drawn from the "major-mergers" catalog (without the "isolated" galaxies), in redshifts $0.5 \leq z \leq 1$ and using dynamical times units. We define them as: "Early" = $[0 < t < T]$, and "Late" = $[T < t]$, and we search for the best time-cut T by comparing the results over the testing-sets to see which time-cut gives the best performance Figure **20**. After that we will be able to determine the time-cut for which "Early" snapshots match **In-Mergers + Post-Mergers** while "Late" snapshots match **No-Mergers**. In order for the comparison between the cases to be fair, we take fixed sizes for the data-sets: 2,200 images for the training-set and 400 images for the testing-set. The data-sets are balanced so the number of "Early" images equals to the number of "Late" images. To make sure that the result of each case actually describes the best score that its time-cut could provide, for each case we repeated the training and testing process 10 times, while resampling the datasets a new at each time (it's a random sampling), and chose the run with the highest score.

Simulation parameters and results:
In each case we trained and tested the network on the simulation's mock images with the following parameters:

- Early [0 – T] vs Late [T+].
- $0.5 \leq z \leq 1$
- Fixed datasets sizes: Training-set: 2,200 ; Testing-set: 400.
- Balancing data-sets only by phase.

Figure **20** shows the results according to the metric of "Recall" (correct labels percentage for **late** snapshots) and "Precision" (purity of **late** snapshots). In Figure **20** we can see that the Recall is high (≥ 0.83) for $T = 0.1, 0.2, 0.3$ dynamical times, with a possible peak at $T = 0.3$ or $0.1\ t_{dyn}$. It tells us that the network does equally well in identifying **late** snapshots for several time-cuts, and only falls dramatically for $T = 0.4\ t_{dyn}$. The Precision however shows a poorer performance for all those values, except a clear peak at $T = 0.3\ t_{dyn}$, where it is also the closest to the Recall (and therefore gives a more balanced result). The bottom line is that $T = 0.3 \pm 0.05\ t_{dyn}$ makes for a visually better time-cut between the "Early" snapshots and the "Late" snapshots, which means that **early** snapshots match **In+Post-Merger**, and **late** snapshots match **No-Mergers** (without the isolated galaxies). The uncertainty of ±0.05 is the half-way to the other time-cuts. The way to interpret it is as following:

1. When we use a time-cut that is lower than $T = 0.3\ t_{dyn}$, we contaminate true "late" images with true "early" images. Then the network learns that whenever it sees a truly "late" image it knows to classify it with the right **late** label, but when it sees a true "early" image, it is unsure of where it belongs and confuses them. Sometimes it labels them as **early** and sometimes as **late**, resulting with a high Recall for the **late** but a poor Precision (a low recall for the **early**).
2. when we use a time-cut that is higher than $T = 0.3\ t_{dyn}$ it's the opposite. we contaminate true "early" images with true "late" images and then the network learns that whenever it sees a true "early" image it knows to classify it with the right **early** label, but when it sees a true "late" image, it guesses. The result is a low Recall for the **late** but possibly with a high Precision (no "early"s are labeled as **late**).
3. when we use the time-cut $T = 0.3\ t_{dyn}$, we get the best division between **early** and **late**, with minimal contamination between the classes. The result is a high Recall **and** a high Precision, with the smallest gap in between them (so most balanced result).

Another thing that we see in Figure **20** is that there is some rise in both the Recall and Precision at $T = 0.1\ t_{dyn}$. We suspect that this is an indication of the middle phase: The "In-Merger" phase, and in section 4.3 we investigate it and constrain its typical duration time.

In Figure **21**, Figure **22** and Figure **23** we can see samples of several images classified by the network as **early** or **late**, with the t/t_{dyn} time cuts: [0 – 0.2 vs 0.2+], [0 – 0.3 vs 0.3+] and [0 – 0.4 vs 0.4+] respectively. We can see the time printed on the images and compare it to the labels, and get an impression on what features led the network to better succeed in which case.

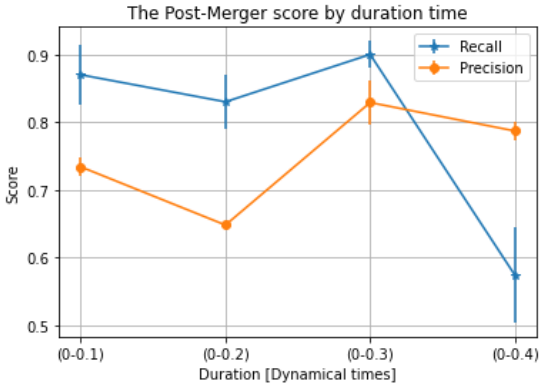


Figure 20: *The score of the **early** [0 – T] vs **late** [T+] classification over different values of T, in dynamical times units, $0.5 \leq z \leq 1$. The scores are the "Recall" (correct labels percentage for **late**) and the "Precision (purity of **late**) where higher is better. We can see that the Recall is high for T = 0.1, 0.2 and 0.3, with a possible peak at T = 0.3 or 0.1. The precision peaks at T=0.3 and closest there to the Recall, indicating that there the results are both high and balanced and $T = 0.3 \pm 0.05\ t_{dyn}$ makes for a better separation time between (In + Post)-Merger and No-Merger. The uncertainty of ± 0.05 is the half-way to the other time-cuts. Notice the rise near (0 – 0.1). We will investigate it.*

<u>The duration of the Major-Merger and the time-gap:</u>

We have found that 0.3 ± 0.05 dynamical times makes for the best definition for the duration of the combined (**inM + postM**) phase, because it gives the best separation between earlier and later snapshots (at least by our models). It defines the merger's end and in order to keep the symmetric definition of the Major-Merger, we define the beginning of the Pre-Merger phase at $t/t_{dyn} = -0.3$ before the first passage ($t = 0$). Later snapshots could be considered as No-Mergers. The sharp change in the score suggests that within a short time ($\pm 0.05\ t_{dyn}$ around $T = 0.3\ t_{dyn}$), the galaxies undergo a significant relaxation. At least this is what visually observed from the simulation by the network. However, these numbers are only true on average, over a large number of galaxies, and we still got not a small confusion between the classes. In order to minimalize this confusion, we also take a time-gap between the end-time of the **postM** class and the start-time of the **no-merger** class, so only snapshots of time $t/t_{dyn} \geq 0.45$ since the first passage ($t = 0$) are taken as **no-merger**s. This value doesn't represent anything physical and was selected via trial and error, as we tried to balance between letting more relaxation time to minimalize the confusion (enlarging the gap), and maintain a large enough number of "late" snapshots ($t/t_{dyn} \geq 0.45$) from the "major-mergers" catalog (shrinking the gap).

* In appendix 3 we check if there is a possible contradiction between these results and section 4.1.1.

<u>A note about Pre-Mergers:</u>

We did try to run similar tests for "Early" Pre-Mergers [$t < -T_{pre}$] and "Late" Pre-Mergers" [$-T_{pre} < t < 0$] (where $T_{pre} > 0$), to see if we can find a less arbitrary definition, but we failed to get any results that support a clear beginning time T_{pre}. It seems that either the **preM** phase is more graduate and any time-cut can be used, or our methods weren't good enough for the task. So we settled with the symmetric value of $T_{pre} = T = 0.3\ t_{dyn}$, which is roughly when the distance between the 2 progenitors is 4 × radius of the main progenitor.

A word of caution – error estimation:

The results that we got here depend heavily on our ability to assess the uncertainty of each measurement (aka the error bars). As we explained before (section 3.3), our error estimation technique accounts for the uncertainty in the noise, different backgrounds sampling and the accuracy of the network in its different predictions. But we can't estimate the uncertainty in the model itself, its training and the architecture. However, what we can do is go around it, and see if and how the different results make sense. In Appendix 4 we explore the learning-rate that we get by using different definitions for the duration of the merger event.

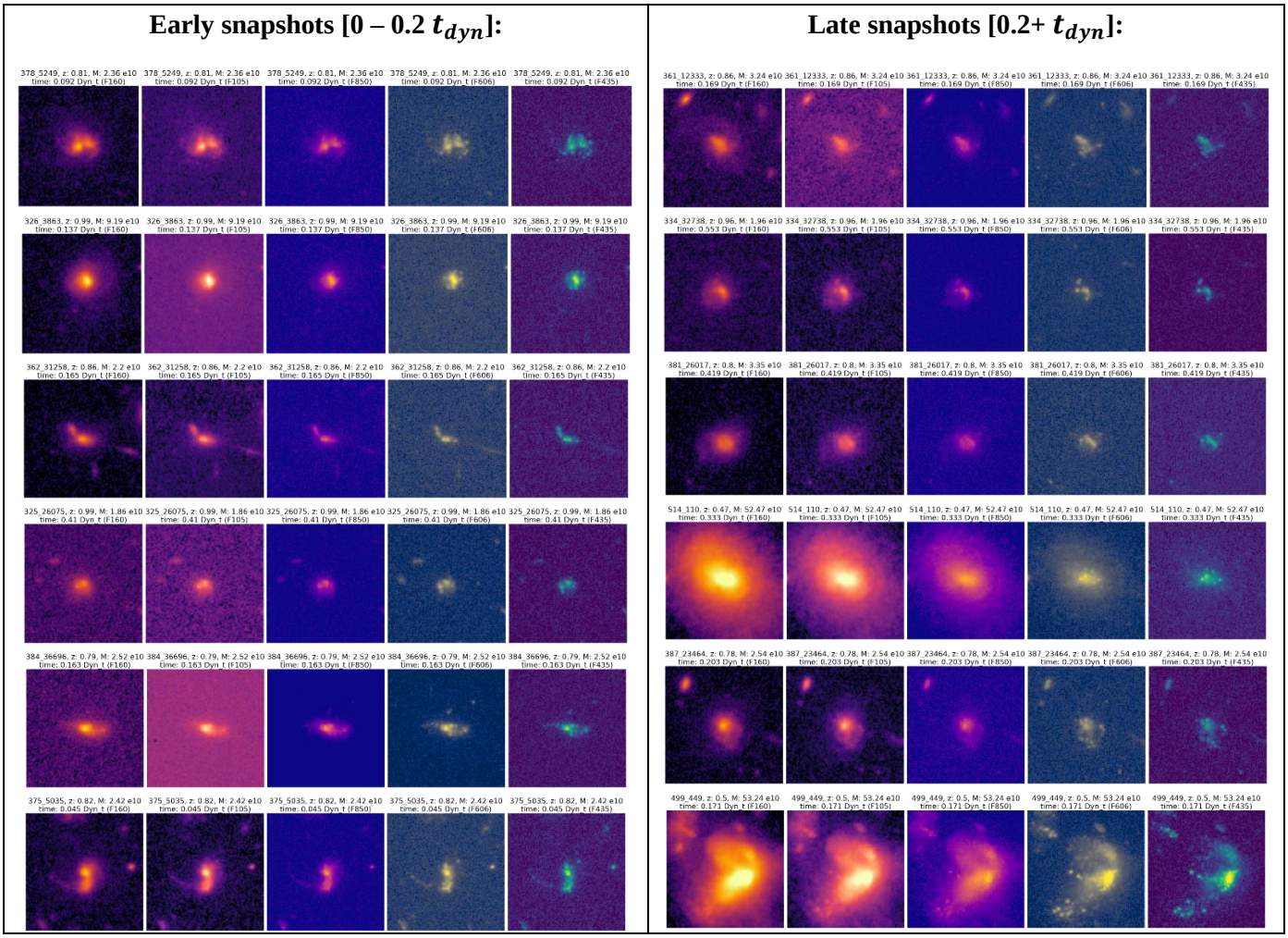


Figure 21: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***early snapshots*** *[0 – 0.2] (lest side) vs* ***late snapshots*** *[0.2+] (right side) classifier in dynamical time units. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass, time since first passage (in t_{dyn}) and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale. We can see on the right side several wrong classifications (according to their time), or just galaxies with distortions like in the left side, even if their time > 0.2 t_{dyn}.*

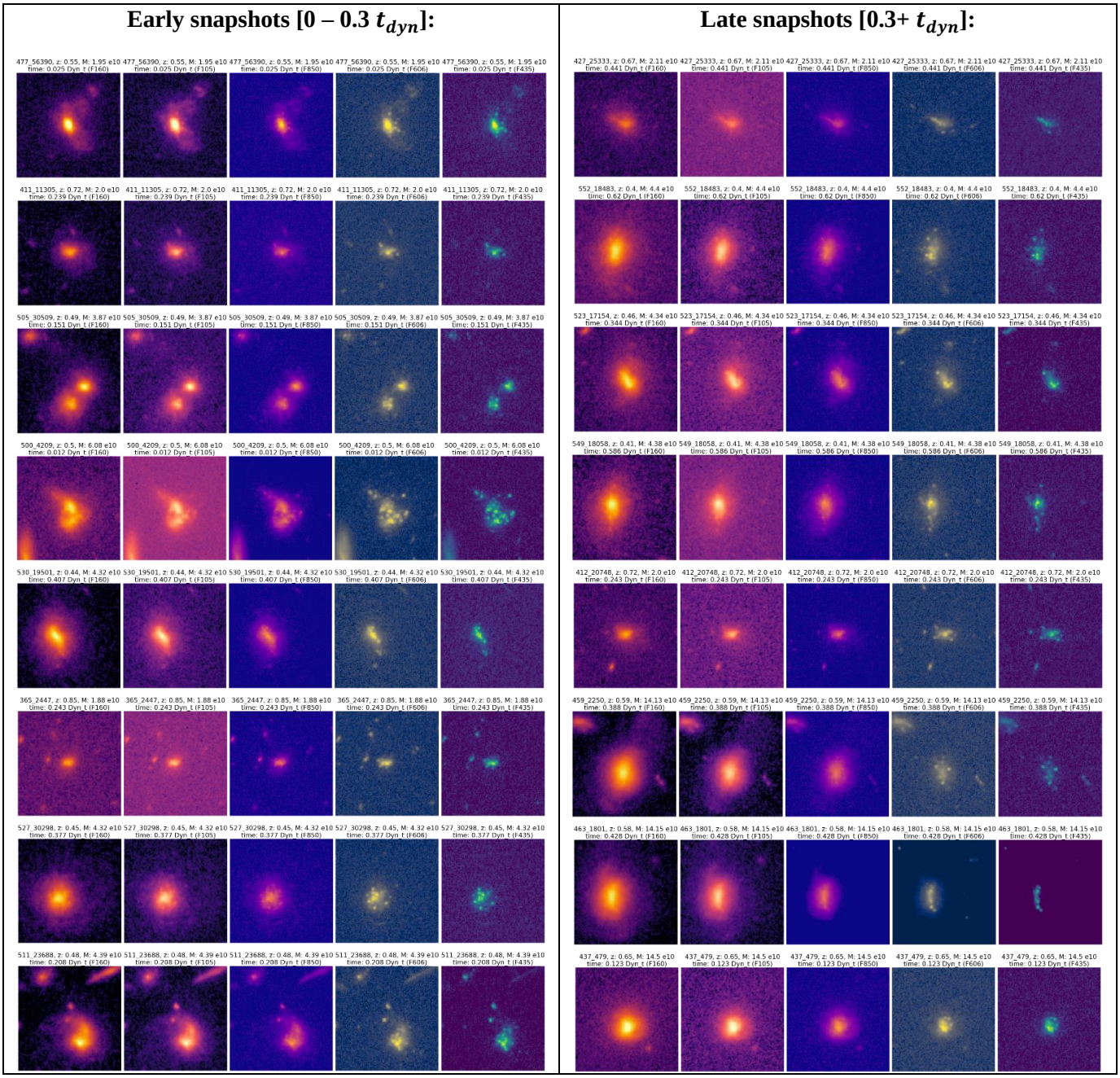


Figure 22: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***early snapshots*** *[0 – 0.3] (lest side) vs* ***late snapshots*** *[0.3+] (right side) classifier in dynamical time units. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass, time since first passage (in t_{dyn}) and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale. Notice the differences between the 2 groups.*

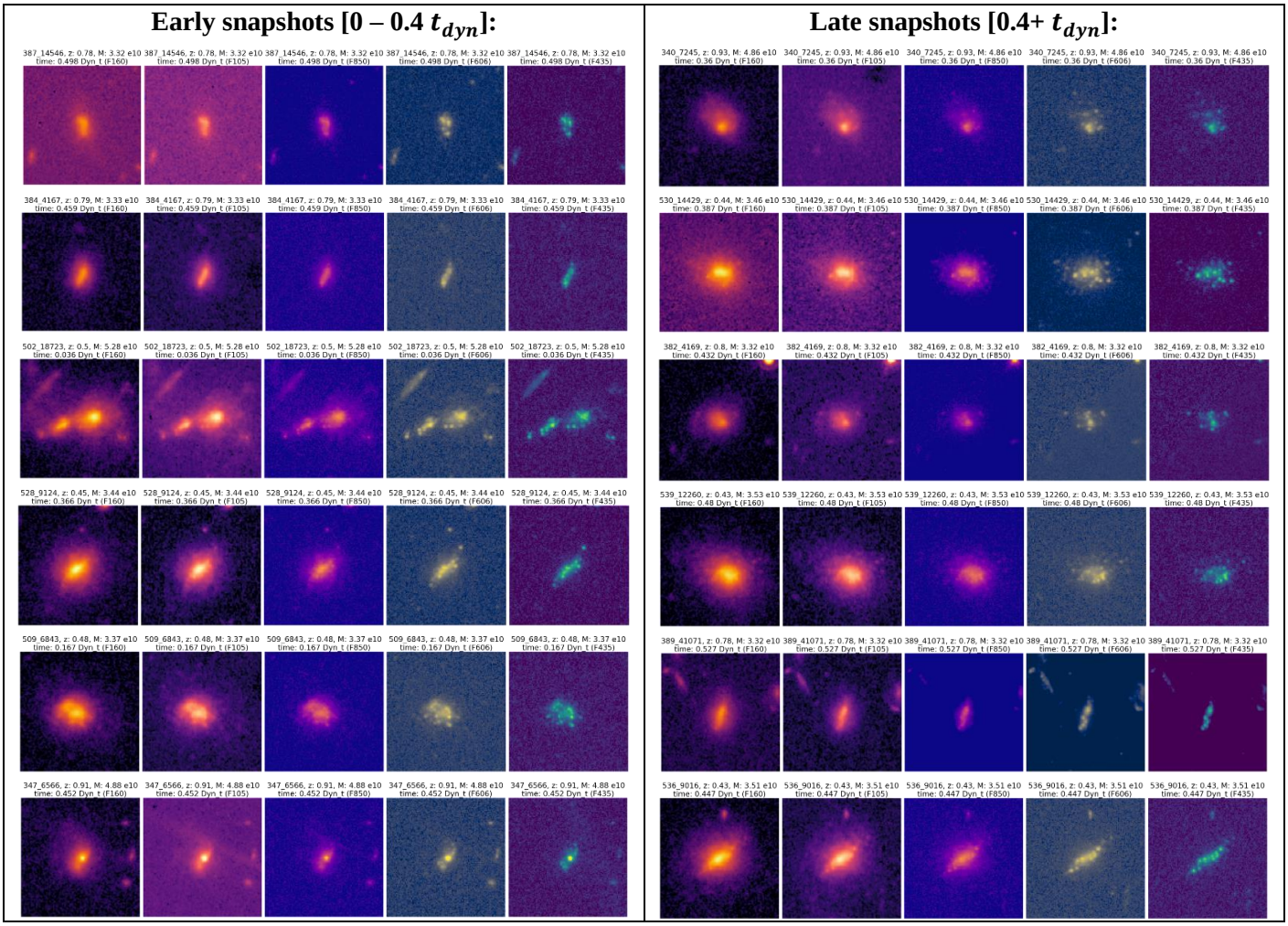


Figure 23: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***early snapshots*** *[0 – 0.4] (lest side) vs* ***late snapshots*** *[0.4+] (right side) classifier in dynamical time units. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass, time since first passage (in t_{dyn}) and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale. We can see on the left side several wrong classifications (according to their time), or just unperturbed galaxies like in the right side, even if their time < 0.4 t_{dyn}.*

4.3. Measuring the "In-Merger" phase

In this part we will explore the middle phase of the merger – the "In-Merger" (**inM**) phase, which we theorize starts at the first passage ($t = 0$ in the catalog) and ends on the final coalescence. In section 4.2, where we looked for a constraint for the duration of the merger (the end time of the (**inM** + **postM**) phase), we saw that there is a rise in the network's performance around $t/t_{dyn} = 0.1$, which might be an indication of another phase transition. In a similar manner, here we train several classifier models to classify **inM** [0 – T] vs **postM** [T – 0.3] with different values of T, in dynamical times units, in redshift $0.5 \leq z \leq 1$. We then compare the metrics' results over the testing-sets to see which definition gives the best performance Figure **27**. In order for the comparison between the cases to be fair, we take fixed sizes for the data-sets: 800 images for the training-set and 200 images for the testing-set. The data-sets are balanced so the number of **inM** images equals to the number of **postM** images. To make sure that the result of each case actually describes the best score that its time-cut could provide, for each case we repeated the training and testing process 10 times, while resampling the datasets a new at each time (it is a random sampling), and chose the run with the highest score. Because this task is harder than classifying **preM** vs (**inM** + **postM**) (evident in the performances), and because we have much less images to work with due to the narrower time windows, we will again do it on images of redshift $z \leq 1$ (to avoid the need to redshift balance the datasets), and we will do it on raw simulation images, without backgrounds, noise or point-spread function (PSF). By removing these effects, we let the network learn from images that are closer to the "ground truth" of the simulation, without noise.

Simulation parameters and results:

We trained and tested the network on the simulation's mock images with the following parameters:

- In-Merger [0 – T] vs Post-Merger [T – 0.3] in dynamical time units.
- $0.5 \leq z \leq 1$
- Fixed datasets sizes: Training-set: 800 ; Testing-set: 200.
- **No PSF no BG** – The idea is to find the duration time, without noise.
- Balancing data-sets only by phase.

Figure **27** shows the results according to the metric of "Recall" (correct labels percentage for **inM**) and "Precision" (purity of **inM**). In Figure **27** we see that the Recall is low (< 0.6) for T < 0.15 t_{dyn} and high (> 0.8) for T > 0.15 t_{dyn}, and the Precision is highest for T = 0.05 t_{dyn} ($\sim$0.8) and around $\sim$0.7 for all the other times. It means that for T < 0.15 t_{dyn} the network straggles to identify the **inM** images, but does better with the **postM** images, especially at T = 0.05 t_{dyn}, and for T > 0.15 the network identifies well the **inM**s, better than it does with the **postM**s. At T = 0.15 t_{dyn} Recall$\sim$Precision which means that the results are the most balanced and unbiased. The interpretation is, like in subsection 4.2.3, that if we take time-cuts that are lower than 0.15 t_{dyn}, the "In-Merger" galaxies fit to both classes, which makes it hard to identify this class and easy to identify the "Post-Merger" images. And if we take time-cuts that are higher than 0.15 t_{dyn} it swaps.

It is an indication that $T = 0.15 \pm 0.025\ t_{dyn}$ makes for the best visually time-cut between the 2 phases. The uncertainty of $\pm 0.025\ t_{dyn}$ is the half-way to the other time-cuts. However, the fact that Recall isn't highest up until T = 0.2 t_{dyn} and the Precision doesn't fall at T > 0.15 t_{dyn} means that in many cases the "In-Merger" phase lasts until 0.2 t_{dyn}. The conclusion is that it is possible that there is

another phase transition at $0.15^{+0.05}_{-0.025}$ dynamical times. That would be the transition from the "In-Merger" phase to the "Post-Merger".

In Figure **24** we can see a sample of several images classified by the network as In-Mergers [0 – 0.05] or Post-Mergers [0.05 – 0.3].

In Figure **25** we see a similar classification for In-Mergers [0 – 0.15] or Post-Mergers [0.15 – 0.3]. In Figure **26** we see a similar classification for In-Mergers [0 – 0.25] or Post-Mergers [0.25 – 0.3]. These figures show examples that support the interpretation given above.

Summary:

We trained and tested a series of classifiers for In-Merger [0 - T] vs Post-Merger [T - 0.3] in dynamical times units, for $0.5 \leq z \leq 1$ *and fixed data-set sizes (800 training images and 200 testing images). We used raw simulated images, without backgrounds, noise or PSF, because we wanted to be closer to the "ground truth" of the simulation and to make it easier for the network that straggled with the very small datasets. We compared the results and discovered that the best time-cut between the 2 phases is* $T = 0.15^{+0.05}_{-0.025}\ t_{dyn}$.

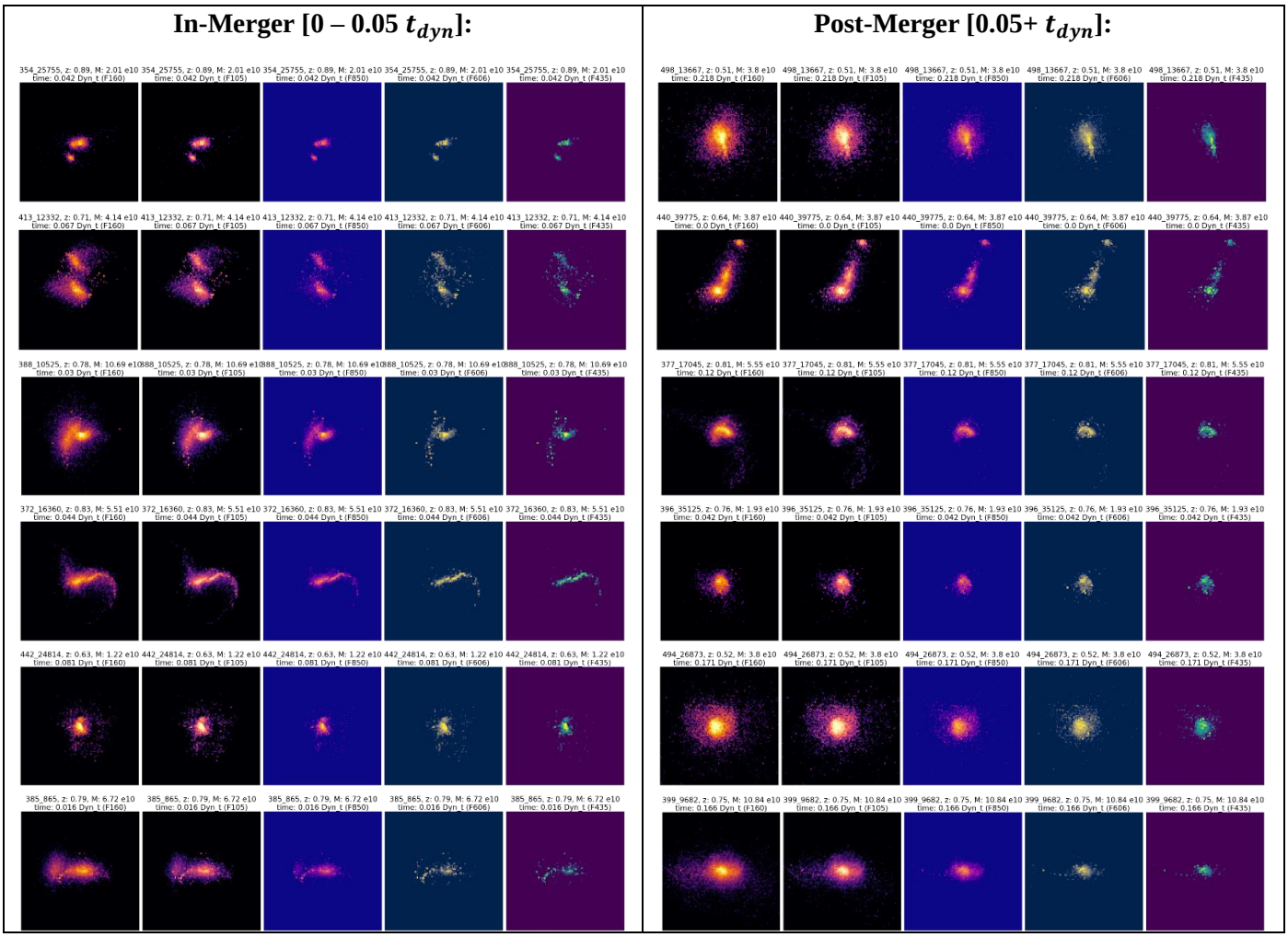


Figure 24: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***inM*** *[0 – 0.05] (lest side) vs* ***postM*** *[0.05+] (right side) classifier in dynamical time units. The ID, redshift, stellar mass, time since first passage (in* t_{dyn}*) and filter of the galaxy are written on top of each image. We can see on the left side that all the images were either classified correctly, or wrongly because they look extremely distorted. On the left side we see a mix of late "calm" galaxies among with very distorted ones that were labeled wrongly.* $t/t_{dyn} = 0.05$ *is probably too low time-cut.*

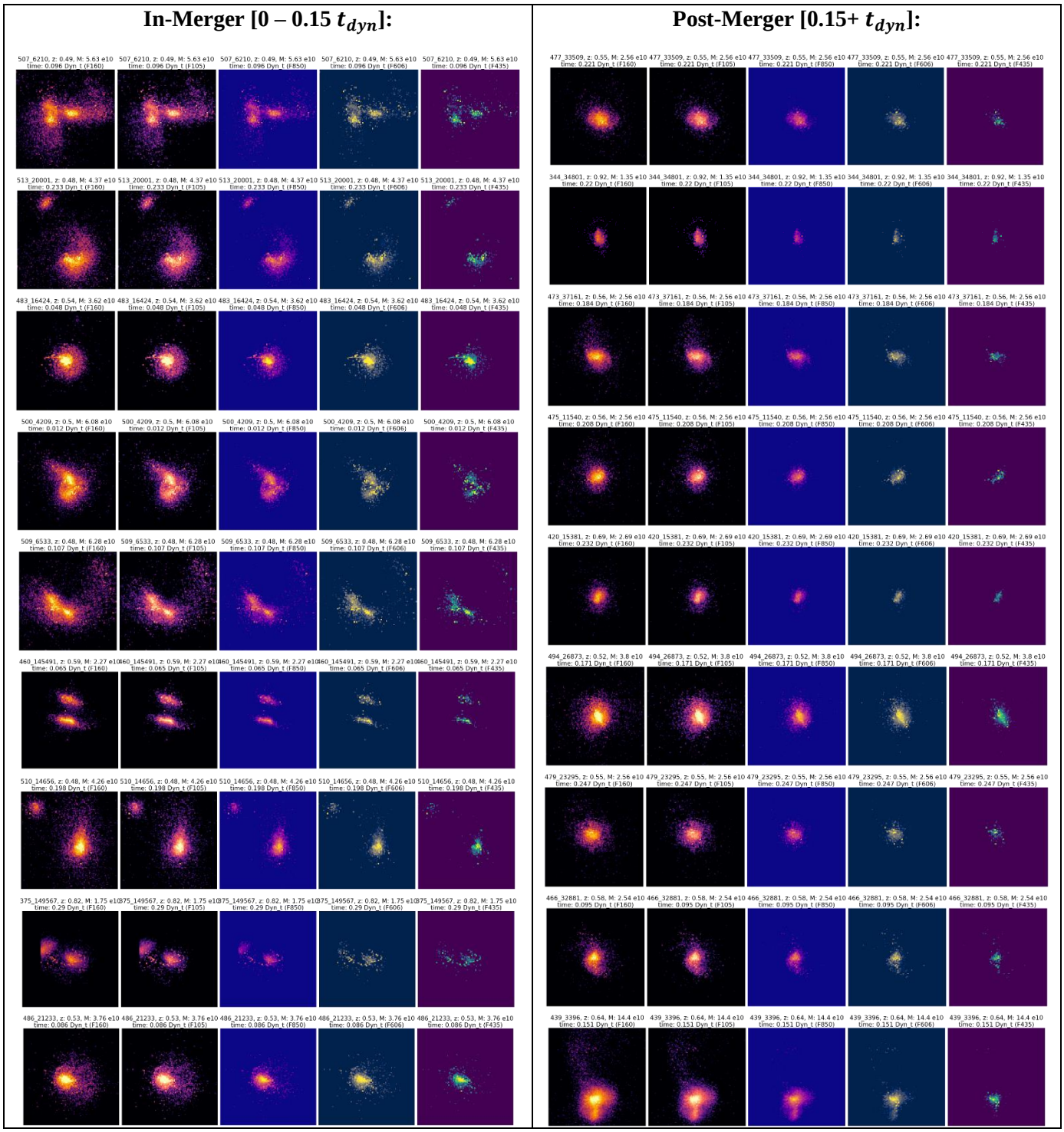


Figure 25: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***inM*** *[0 – 0.15] (lest side) vs* ***postM*** *[0.15+] (right side) classifier in dynamical time units. The ID, redshift, stellar mass, time since first passage (in t_{dyn}) and filter of the galaxy are written on top of each image. We can see on the left side that almost all the images have several galaxies or extreme distortions, including the wrongly labeled images according to their time. On the right side we see only 1-galaxy images with only mild distortions. The classification isn't perfect so probably on a time-range around $t/t_{dyn} = 0.15$ many galaxies transit from the "In-Merger" phase into the "Post-Merger" phase.*

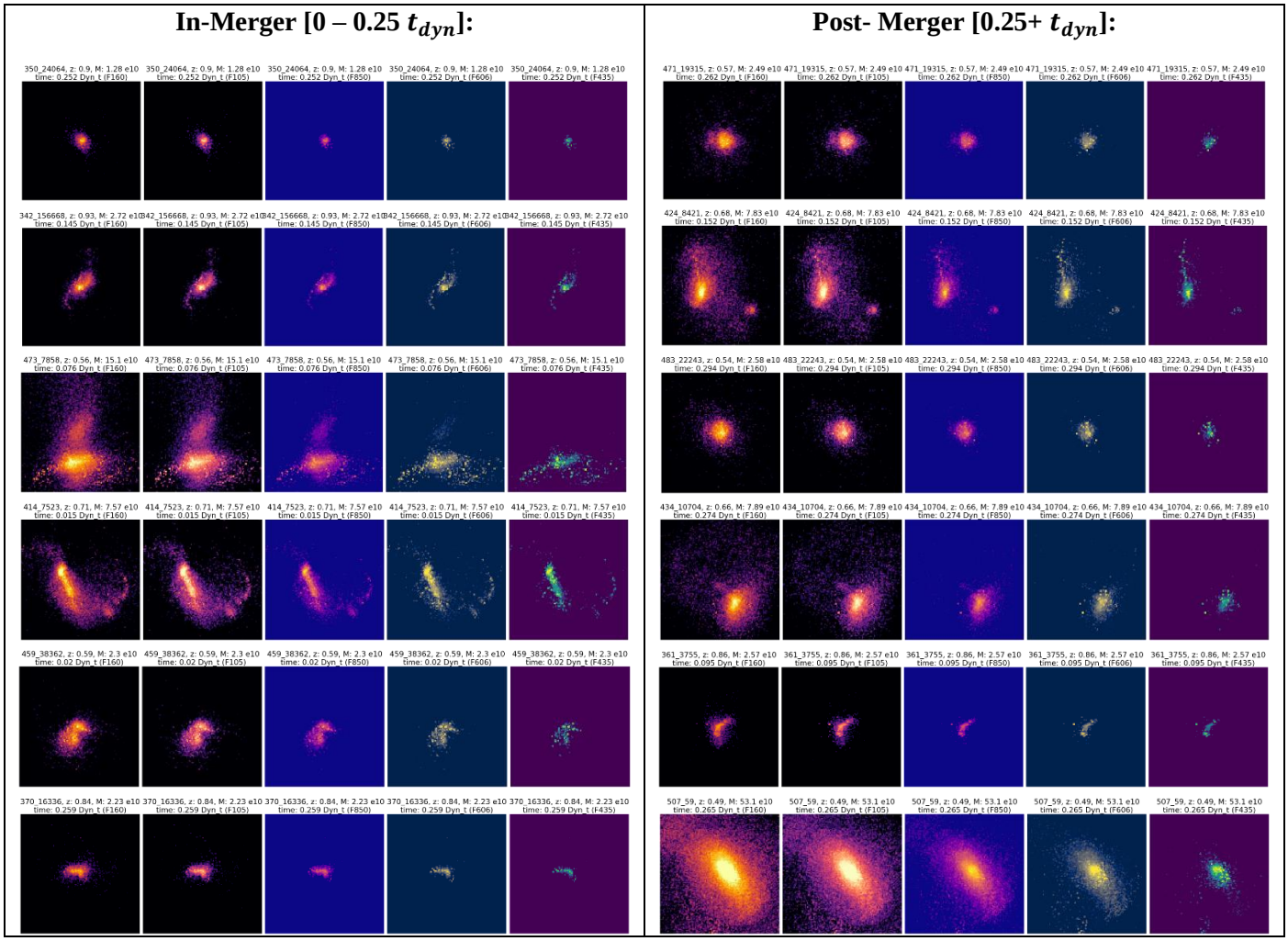


Figure 26: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***inM*** *[0 – 0.25] (lest side) vs* ***postM*** *[0.25+] (right side) classifier in dynamical time units. The ID, redshift, stellar mass, time since first passage (in t_{dyn}) and filter of the galaxy are written on top of each image. We can see on the left side a clear mix of regular and calm galaxies, but mostly correctly classified. Surprisingly, on the right side we see a few very early and distorted galaxies. This confusion probably means that $t/t_{dyn} = 0.25$ is a too high time-cut.*

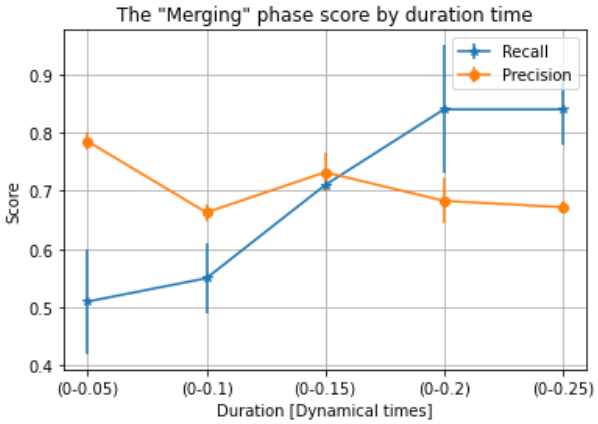


Figure 27: *The score of the* ***inM*** *[0 – T] vs* ***postM*** *[T – 0.3] classification over different values of T, in dynamical times units, $0.5 \leq z \leq 1$. The scores are the "Recall" (correct labels percentage for* ***inM****) and the "Precision" (purity of* ***inM****) where higher is better. We can see that the Recall is low for $T < 0.15t_{dyn}$ and high for $T > 0.15t_{dyn}$, and the Precision is highest (~0.8) for $T = 0.05t_{dyn}$ and around ~0.7 for all the other times. At $T = 0.15\ t_{dyn}$ the results are the most balanced (Recall~Precision). It's an indication that $T = 0.15 \pm 0.025\ t_{dyn}$ makes for the best visually time-cut between the 2 phases. The fact that at $T = 0.15t_{dyn}$ Recall is not highest and Precision doesn't fall means that in many cases the* ***inM*** *phase lasts until $T = 0.2\ t_{dyn}$. So, it is possible that there is another phase transition at $0.15^{+0.05}_{-0.025}\ t_{dyn}$ dynamical times.*

4.4. Our Merger-Fraction and Merger-Rate

With the knowledge we have collected so far, we can now train a new **merger** vs **no-merger** classifier, with observation time windows that are fixed fraction of the DM halo's dynamical time. It means that the duration of the merger and its phases vary with redshift. We use this classifier on real CANDELS images and calculate the new Merger-Fraction and Merger-Rate, as a function of the redshift. To complete the saga we also train a classifier for **preM** vs (**inM** + **postM**), again with the new observation time window in dynamical times units.

4.4.1. Mergers vs No-Mergers - Measuring merger-fraction and merger-rate

We train a classifier for Major-Mergers vs No-Mergers, and produce the merger-fraction and merger-rate as a function of redshift.

Simulation parameters and results:

We trained and tested the network on the simulation's mock images with the following parameters:

- Mergers [-0.3 – 0.3] vs no-mergers [0.45+] in dynamical time units.
- Training-set size: 11,500 images.
- Testing-set size: 1,500 images.
- Among the **no-merger** images, 15% (885/5,750) in the training-set and 16% (117/750) in the testing-set, are "late" images from the "major-mergers" catalog.

Redshift	Number of images	phase	Number of images
$0.4 \leq z < 1$	4,138	Pre-Merger	2,875
$1 \leq z < 1.5$	3,784	inM + postM	2,875
$1.5 \leq z < 2$	3,578	No-Merger	5,750

Table 10: *The distribution of the training-set's mock images from the simulation after they were balanced by redshift bins (**left**) and phase (**right**). The case is of **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] in dynamical time units. Notice that the number of **no-merger** images equals to the number of **preM** + **inM** + **postM** images. However, there are slightly more images in lower redshifts than in higher redshifts (up to 14% difference at maximum). The high score means that this imbalance isn't impactful.*

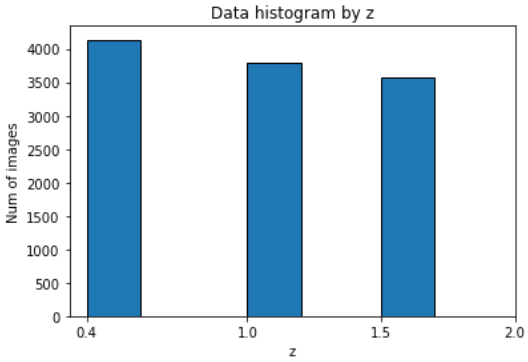


Figure 28: *A histogram the shows the distribution of the training-set's mock images from the simulation by redshift bins, after balancing. The case is of **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] in dynamical time units.*

Table **10** and Figure **28** show the number of images in the training-set, and their distribution over redshifts and phases. We can see that they are perfectly balanced with respect to phase (number of **no-merger** = **merger** and the numbers of **preM** and (**inM** + **postM**) are equal), but with respect to redshifts there's a difference of up to 14% between the largest bin and the smallest one. The reason for this is that some of the **no-merger** images, specifically the "late" snapshots ($t/t_{dyn} > 0.45$) that were drawn from the "major-mergers" catalog (see section 2.2.1), are a very small group so we had to keep as many of

them as we could, while compromising a little on the balancing by redshift bins. The high score means that this imbalance isn't impactful.

Table **11** shows the results according to the 2 interchangeable metrics: "Accuracy" and "Balance/difference-rate", "Precision" and "Recall". We can see that all the results are above 90%, with a very minimal error, which means the network learned very successfully. Figure **29** shows the confusion matrix, which visualizes the results with the same coding as before. The results are highly balanced with a small bias (0.015 ± 0.002) toward the **no-merger** class, and the small error is the result of the fact that the network is very confident in the predictions (see Figure **30**, so even after adding/subtracting the prediction error for each prediction (which is also small because the accuracy is very high), most of the galaxies don't switch classes ($> 50\% \Rightarrow$ **merger**; $< 50\% \Rightarrow$ **no-merger**).

Figure **31** shows a sample of images classified by the network. From there we can see that **merger** galaxies are characterized by visible tidal features or other distorted morphology, like asymmetries between the shape of the core and the rest of the galaxy, or a pair of galaxies that both of them are bright in all the bands (and in many cases they also show distorted morphology). In contrast, **no-merger** galaxies seem more "nice", symmetric and they tend to be fainter in the F606 (blue) and F435 (purple) bands than in the F160 (red) and F105 (yellow) bands, especially in the outskirts, as oppose to **merger** galaxies. These characterizations aren't always true though, which cause the confusions.

Metric	Score
Mean prediction error	0.0078
Accuracy	0.908 ± 0.008
Balance-rate	0.985 ± 0.008
Precision	0.903 ± 0.002
Recall	0.916 ± 0.005

Table 11: *The score of the **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] classification in dynamical time units, over the simulation's testing-set. The Metrics' meaning are explained in section 3.3 and the errors represent 1σ prediction-error estimation added to / subtracted from the predictions.*

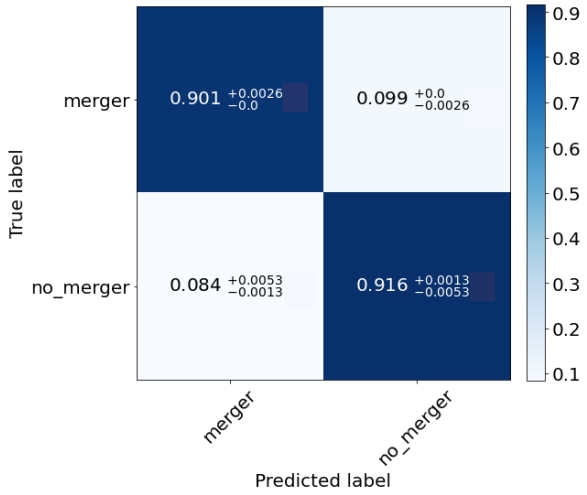


Figure 29: *Normalized confusion matrix for the **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] classification results in dynamical time units, over the simulation's testing-set. Each row represents the true labels of the relevant class, and each column represents the predicted label. The results are highly balanced with a small bias (0.015) toward the **no-merger** class. The small errors mean a high confidence for the predictions.*

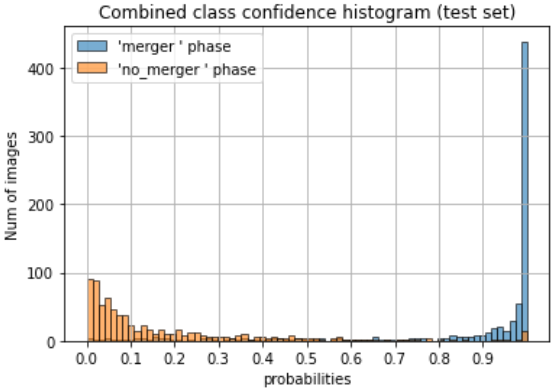


Figure 30: *Probability distribution of the prediction results of the **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] classification in dynamical time units, over the simulation's testing-set. The classifier is more confident in the **merger** class than in the **no-merger** one, although according to the confusion matrix the accuracy of the **no-merger** class is higher.*

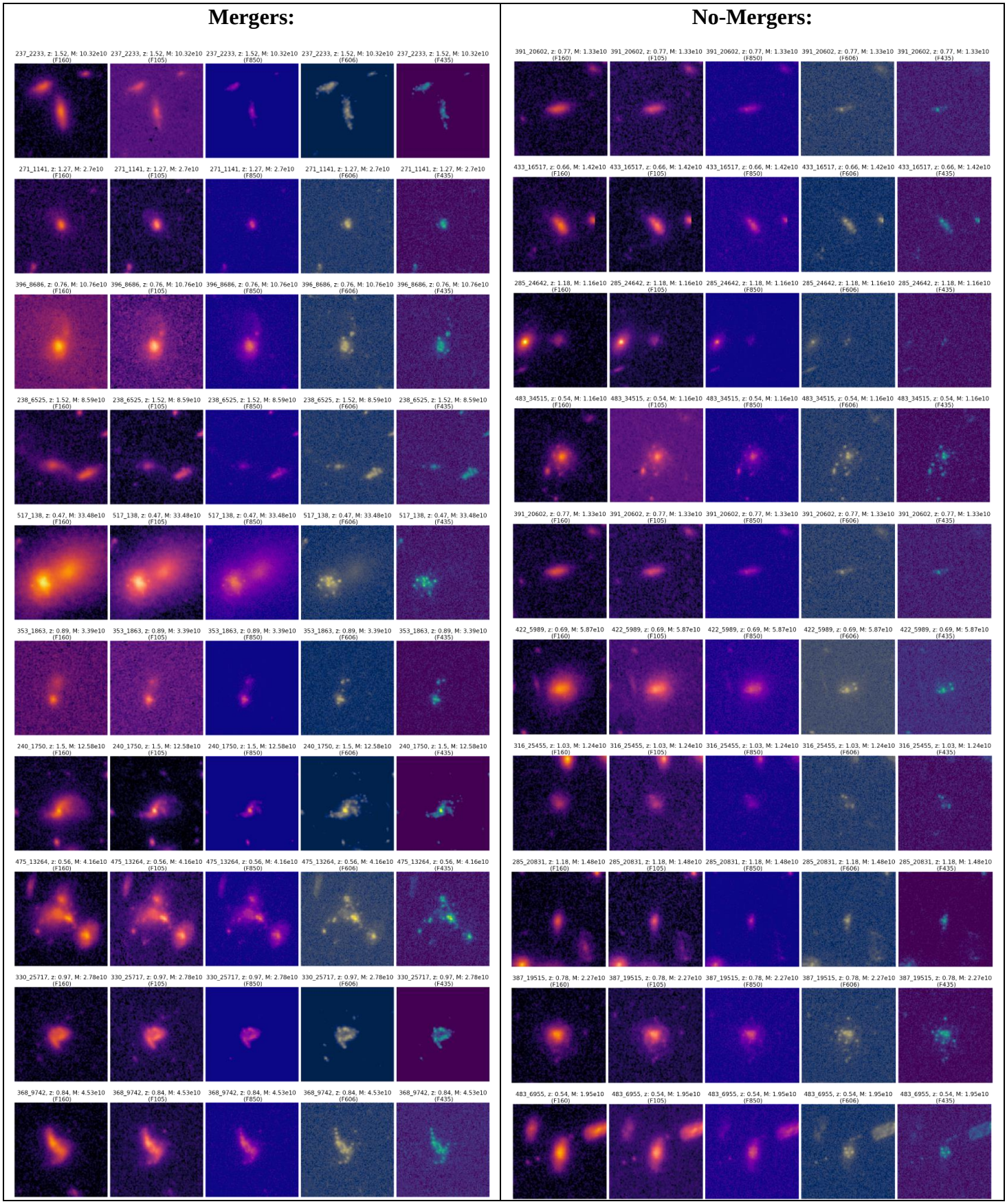


Figure 31: *A sample of mock-images drawn from the simulation's testing-set, labeled as they were classified by the* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.45+] classifier in dynamical time units. Mergers on the left side, No-Mergers on the right side. In each side, each row has 5 images of the same galaxy in different filters. The ID, redshift, stellar mass and filter of the galaxy are written on top of each image. All the images' colormaps are set to the same scale.*

Summary:
*We trained a model to classify **mergers** [-0.3 – 0.3] vs **no-mergers** [0.45+] in dynamical times units and tested it with very high accuracy (0.908 ± 0.008), high balance (0.985 ± 0.008) and high confidence. We measured the error range by applying the prediction-error to every prediction result before projecting it again to one of the classes, and almost none of them changed classes. **merger** galaxies have distorted morphology, or a pair of bright galaxies in all the bands (in many cases they're also distorted). **no-merger** galaxies look "nice", symmetric and fainter in the bluer bands than in redder ones, especially in the outskirts.*

Real CANDELS results:

We then used the new trained **merger** vs **no-merger** classifier on real CANDELS galaxies, and compared it to our previous classifier from section 4.1.1 (with absolute time units).

Table **12** shows the number of real CANDELS galaxies and their percentage that were classified in each class (**merger** and **no-merger**). Again, high confidence in the predictions with a small error estimation leads to a small number of galaxies that switch classes (2/49 in our case).

We got only 49 galaxies that were classified as mergers, while in section 4.1 (absolute times units) we got 73 mergers. The difference in the numbers makes sense, because in our real CANDELS catalog there are more images in $1 < z \leq 2$ than in $0.5 \leq z \leq 1$, and in higher redshifts the observation time window of the current model that uses dynamical times is smaller than in the 4.1.1-model that uses constant absolute times, so we expect to find less galaxies there and in total.

Phase	Total amount	Percentage
Mergers	49^{+1}_{-1}	$0.086^{+0.002}_{-0.002}$
No-Mergers	524^{+1}_{-1}	$0.914^{+0.002}_{-0.002}$

***Table 12:** The total number of real CANDELS galaxies and their percentage that were classified for each phase by the **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] classifier in dynamical time units. The errors are estimated by adding/subtracting the 1σ prediction-errors from the simulation, to/from the prediction results of the real CANDELS, and then casting each prediction to either "Merger" (> 0.5) or "No-Merger" (< 0.5) phase. High confident predictions with small errors lead to barely any change in the final phase.*

We sorted all the galaxies into several redshift bins and calculated the merger-fraction and merger-rate within each redshift, as well as the uncertainty that comes from the galaxies that switched labels. The merger-rate is calculated by dividing the merger-fraction with the observation time ($0.3\ t_{dyn}$).

In Figure **32** we plot our new merger-fraction and in Figure **33** we plot our new merger-rate (both with t_{dyn}), as a function of redshift (red dots with the uncertainty as the error-bars) and add the best fit (orange line) with 1σ error of the fit (orange shaded area). In each figure, the left panel is in a half logarithmic scale and the right panel is in a full logarithmic scale. In each figure, the formulas of our new (with t_{dyn}) best merger-fraction fit and merger-rate fit are given on top the panels. For comparison, in the same graphs we also plot our previous merger-fraction and merger-rate (green stars) that we got in section 4.1.1 with absolute time units, best fit (purple line) and 1σ error (purple shaded area).

We can see that the new merger-fraction are higher in low redshifts and lower in high redshifts, as expected by defining the observation time as a fraction of the halo's dynamical time. The observation time is longer at low redshifts and shorter at higher redshifts. The merger-rate however, turned out to be the same, with both formula coefficients (the scale and the power law) are within the error ranges of each case. The merger-rate in equation (21) does not fully match the theoretical estimation we gave in equation (5), but the power-law is very similar (within 1σ error we got minimum 2.6 and equation (5

) 2.5). In 2σ we do match. We didn't expect a full match because equation (5) estimates the full halo accretion rate and we only look at galaxy major-mergers of a limited mass-cut ($M_* \geq 10^{10} M_\odot$). Our merger-rate at $z = 2$ (0.36 – 2.68 Gyr^{-1}) also matches the merger-rate from equation (6) at the same redshift (0.45 Gyr^{-1}).

At first glans the matching of the results in Figure **33** seems surprising, because in section 4.2.3 we saw that taking different time-cuts for the duration of the merger can seriously damage the performances, which lead us to the conclusion that after 0.3 dynamical times the galaxies do look distinctly different. Therefore, we didn't expect the merger-rate to be the same whether we define a fixed or a redshift-depended observation time. But as explained in appendix 3, by using a wide gap of 0.2 Gys between the **merger** [-0.3 – 0.3 Gys] and **no-merger** [0.5+ Gys] classes, we ensured that all the **no-merger** galaxies were "above" 0.3 dynamical times and most of the **merger** galaxies (except some of the high redshift galaxies) were "below" 0.3 dynamical times. And indeed the performances were high in both runs.

Another validation for the importance of the duration of the merger event in dynamical times and the specific value of 0.3 t_{dyn}, will be to check if we also get or not the same merger-rate even with different time-cuts (in dynamical times units). We show it in Appendix 4.

$$MF(z) = (0.013 \pm 0.011) \cdot (1+z)^{2.31 \pm 0.88} \qquad (20)$$

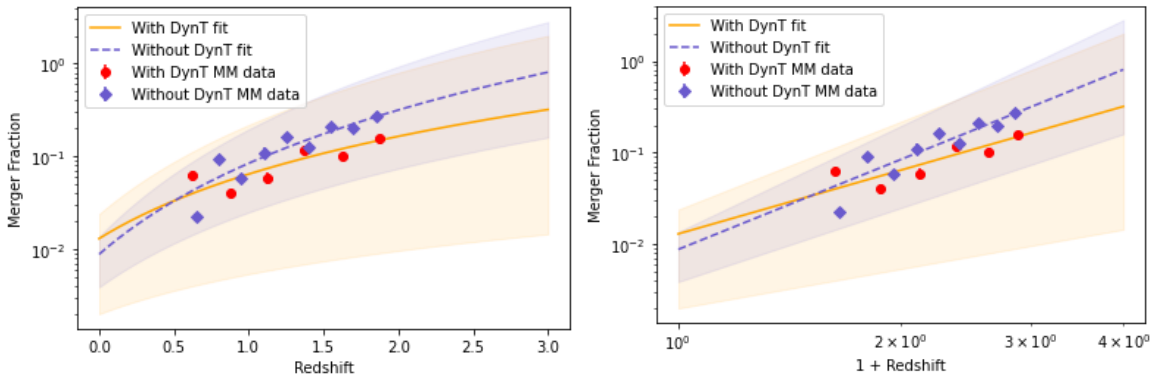


Figure 32: *Left: Merger-Fraction vs redshift in a half-log scale. Right: Merger-Fraction vs (1+redshift) in a log-log scale. The case is of* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.45+] in dynamical time units, over real CANDELS images. The lines are the best fit and the shaded areas are the 1-σ error-range of the fits. We add the previous results that we got in section 4.1.1 (with Gys) in purple colors for comparison. We can see that the new Merger-Fraction are higher at low z and lower at high z, as expected due to the varying observation window.*

$$MR(z) = (0.011 \pm 0.009) \cdot (1+z)^{3.55 \pm 0.91} [Gyr^{-1}] \qquad (21)$$

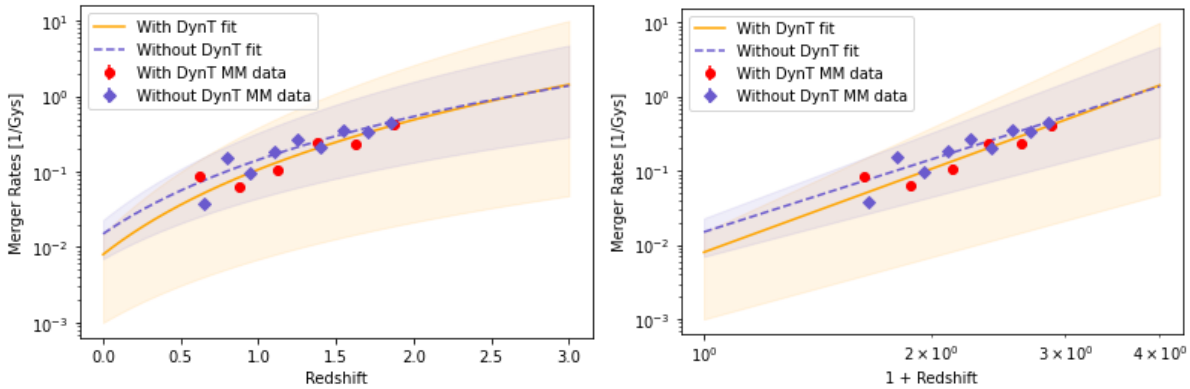


Figure 33: *Left: Merger-Rate vs redshift in a half-log scale. Right: Merger-Rate vs (1+redshift) in a log-log scale. The case is of* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.45+] in dynamical time units, over real CANDELS images. The lines are the best fit and the shaded areas are the 1-σ error-range of the fits. We add the previous results that we got in section 4.1.1 (with Gys) in purple colors for comparison. We can see that there's a good agreement, which suggests that varying the observation time window at low and high z, doesn't confuse the network.*

Summary:

We applied the merger vs no-merger classifier on real CANDELS images, resulting in 49^{+1}_{-1} / 573 *mergers (less than in section* 4.1.1 *but it is expected). We sorted them into redshift bins and calculated the merger-fraction and merger-rate within each bin. The merger-rate match the results we got when using fixed absolute times, which is explained in appendix 3. The power law that we got (minimum 2.6) is also similar, but not fully match the theoretical value (2.5) in equation (5) (does match if we take 2σ error) and at* $z\sim2$ *our merger-rate's magnitude (0.36 – 2.68* Gyr^{-1}*) matches the theoretical value in equation (* 6 *) (0.45* Gyr^{-1}*). This is the most robust and unbiased measurement of the merger-rate up to z=2 that we know of, using machine learning, objective labeling and objective time-durations that scales as the DM halo's dynamical time.*

4.4.2. Pre-Mergers vs (In-Mergers + Post-Mergers)

We train a classifier for Pre-Mergers vs the combined (In-Mergers + Post-Mergers) phases with a redshift-depended duration, that is proportionate to the halo's dynamical time, for the galaxies that were classified as mergers in section 4.4.1.

Simulation parameters and results:
We trained and tested the network on the simulation's mock images with the following parameters:

- Pre-Mergers [-0.3 – 0] vs (In-Mergers + Post-Mergers) [0 – 0.3] in dynamical time units.
- Training-set size: 9,120 images.
- Testing-set size: 906 images.

Table (13) shows the results according to the 2 interchangeable metrics: "Accuracy" and "Balance/difference-rate", "Precision" and "Recall". We can see that the performance is not as good and balanced as in the **merger** vs **no-merger** case, indicating that classifying between inner phases of a merger is a harder task, but still good enough to work with. Figure **34** shows the confusion matrix, which visualizes the results in a clearer way. The results have a confusion of 0.177 – 0.206 between the classes, and are balanced with a bias of 0.03 ± 0.02 toward the **preM** class. The error is of the scale of 1%-5% of the value, which is an indication for the confidence in the predictions (see Figure **35**).

Figure **37** shows a sample of images classified by the network. From there we can see that roughly, the network sorts out the relevant galaxies from the backgrounds by taking only galaxies which are more dominant and equally bright in all the bands. Then if it has 1 galaxy it is classified as a (**inM** + **postM**) and if it has 2 it is a **preM**. Confusion happens mostly when a **preM** image shows 1 galaxy (maybe the other one hides behind the first one?), or a (**inM** + **postM**) image shows several galaxies (usually within a short time after the merging. Probably the **inM** phase) or when there are many objects that are equally bright and appear in all the bands. These characterizations aren't always true though.

Metric	Score
Mean prediction error	0.028
Accuracy	0.808 ± 0.028
Balance-rate	1.029 ± 0.023
Precision	0.818 ± 0.008
Recall	0.794 ± 0.014

Table 13: *The score of the **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in dynamical time units, over the simulation's testing-set. The Metrics' meaning are explained in section 3.3. We can see a balance rate > 1, which means that there's a bias towards **preM** (< 1 is a bias towards (**inM** + **postM**) phase).*

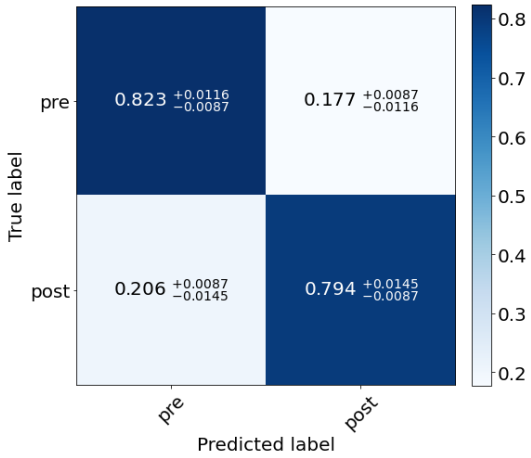


Figure 34: *Normalized confusion matrix for the **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in dynamical time units, over the simulation's testing-set. Each row represents the true labels of the relevant class, and each column represents the predicted label. There's a small bias of 0.029 toward the **preM** phase.*

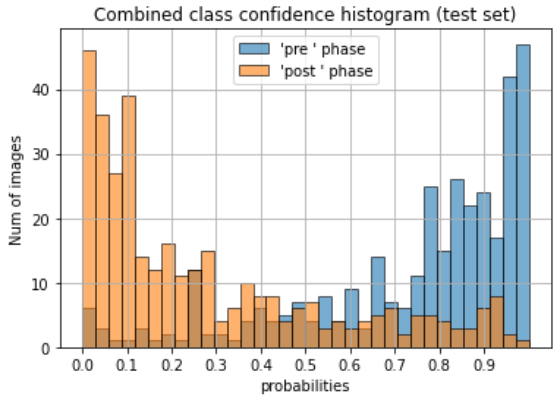


Figure 35: *Probability distribution of the prediction results of the **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] classification in dynamical time units, over the simulation's testing-set. There are not a few cases of confusion, indicating that the 2 phases sometimes look alike. Could be around the merging at $t = 0$.*

Real CANDELS results:

We used the trained **preM** vs (**inM** + **postM**) classifier model on real CANDELS galaxies, and compared it to our results in section 4.1 (with absolute time units). We took only the galaxies that were previously classified as Mergers by the **merger** vs **no-merger** classifier (part 4.4.1) (dynamical time units), classified them into **preM**s or (**inM** + **postM**)s and calculated the ratio of our **preM** / (**inM** + **postM**) within different redshifts and in total. Since we defined the duration time of each phase to be symmetric (0.3 t_{dyn}), we expect a ratio of $PreM/(inM + postM) \rightarrow 1$

Table **14** shows the number of real CANDELS **merger** galaxies and their percentage that were classified in each class (**preM** and **inM** + **postM**), with 2σ prediction-errors that leads to 6/49 galaxies that change their labels. We see a very balanced result with a ratio of just $preM/(inM + postM)\,\big|_{t_{dyn}} = 1.04^{+0.41}_{-0.16}$, which is within the same range of what we got with fixed absolute time unit (subsection 4.1.2): $1.09^{+0.34}_{-0.17}$ (both error ranges are of 2σ).

Phase	Total amount	Percentage
Pre-Mergers	25^{+4}_{-2}	$0.51^{+0.082}_{-0.041}$
In+Post-Mergers	24^{+2}_{-4}	$0.49^{+0.041}_{-0.082}$

Table 14: *The total number of real CANDELS galaxies and their percentage that were classified for each phase by the **PreM** [-0.3 – 0] vs **PostM** [0 – 0.3] classifier in Gys units (not dynamical time). We only considered galaxies that were classified as "Mergers" by {4.4.1}. The errors are estimated by adding/subtracting the 2σ prediction-errors from the simulation, to/from the prediction results of the real CANDELS. Notice that the results are very balanced, with a tiny edge to the **PreM** phase, like we got in the simulation's score.*

We sorted the galaxies into several redshift bins. In Figure **36** we plot the fraction of galaxies that were classified as **preM** (blue down-rectangles) and the fraction of galaxies that were classified as (**inM** + **postM**) (orange up-rectangles), within each redshift bin, normalized such that their sum is always 1. The left panel is of our results with a redshift depended observation window (dynamical time units), and the right panel is of our previous results with a fixed observation time window (Gys units), for comparison. The error bars in both panels are the result of the 2σ uncertainty in the prediction, as described above.

In Figure **36** we can see a comparison between our results with dynamical times (left) and without (right). In both panels in about half of the redshifts the majority of the galaxies are **preM** and in the other half it is (**inM** + **postM**), and it seems randomly ordered. And the gaps on the left panel are at maximum about $\pm 0.1 t_{dyn}$, which looks even better than what in the right panel. We think that it is a satisfying result that shows that our network classifies successfully **preM** vs (**inM** + **postM**) also, and perhaps even better, when using dynamical times for the observation window. Especially if we remember that we have about just 10 images on average for each bin (so ~5 for **preM** and ~5 for (**inM** + **postM**)). However, one might claim that these results also support a random guessing, but the results from the simulation tell us that the network **does not** try a random guessing strategy. Also, by looking at the real CANDELS images in Figure **37** it doesn't look like a random guessing.

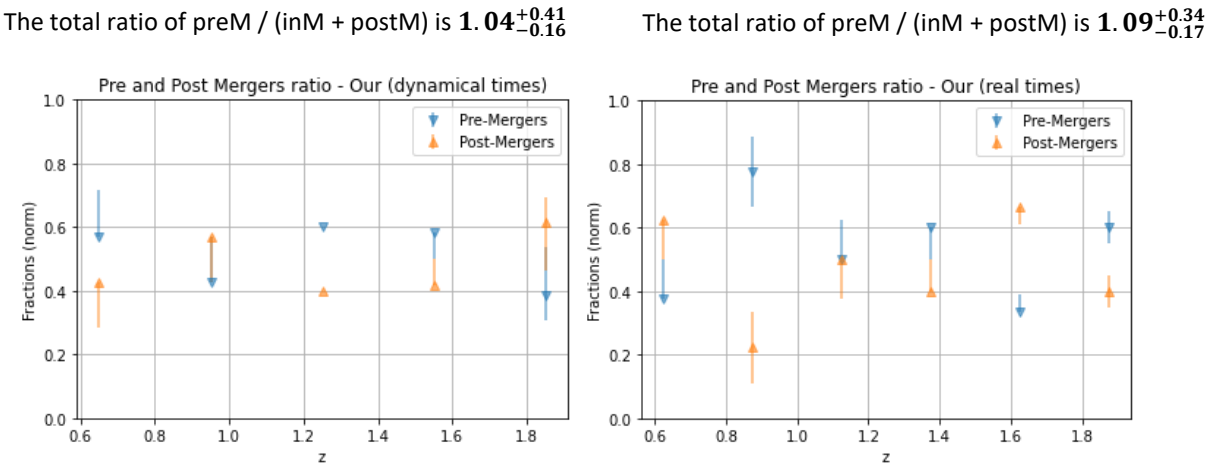


Figure 36: *The normalized fraction of **preM** and (**inM** + **postM**) predictions in different redshift bins in our work, with dynamical time units (left) and with absolute time (Gys) units (right), among the real CANDELS galaxies that were classified as "Mergers" by {4.4.1} and {4.1.1} respectively. The errors in both panels represent 2σ over the predictions. The facts that our results are close in most of the bins, that they flip from **preM** dominance to (**inM** + **postM**) dominance randomly and that the ratio **preM** / (**inM** + **postM**) is close to 1, are indications of a well-balanced classification. Even more so in the dynamical time units case (left).*

<u>*Summary:*</u>

*We trained a model to classify **preM** [-0.3 – 0] vs (**inM** + **postM**) [0 – 0.3] in dynamical times units and tested it with accuracy of 0.808 ± 0.028, balance 1.029 ± 0.023 and good confidence. The performance are not as good as in the case of **merger** vs **no-merger** classification (subsection 4.4.1), but are better than in the case with absolute times (subsection 4.1.2). Then we applied the model to real CANDELS galaxies that were previously classified as mergers. There isn't a visible bias in different redshifts and* 25^{+4}_{-2} */ 49 were classified as **preM**, which is a very balanced ratio* ($1.04^{+0.41}_{-0.16}$) *that is similar to what we got using real absolute times* ($1.09^{+0.34}_{-0.17}$) (using 2σ error estimation). *According to images we see that first the network sorts out galaxies that are distorted or bright in all the bands, then if there's only 1 it's usually (**inM** + **postM**) and if more it's **preM**.*

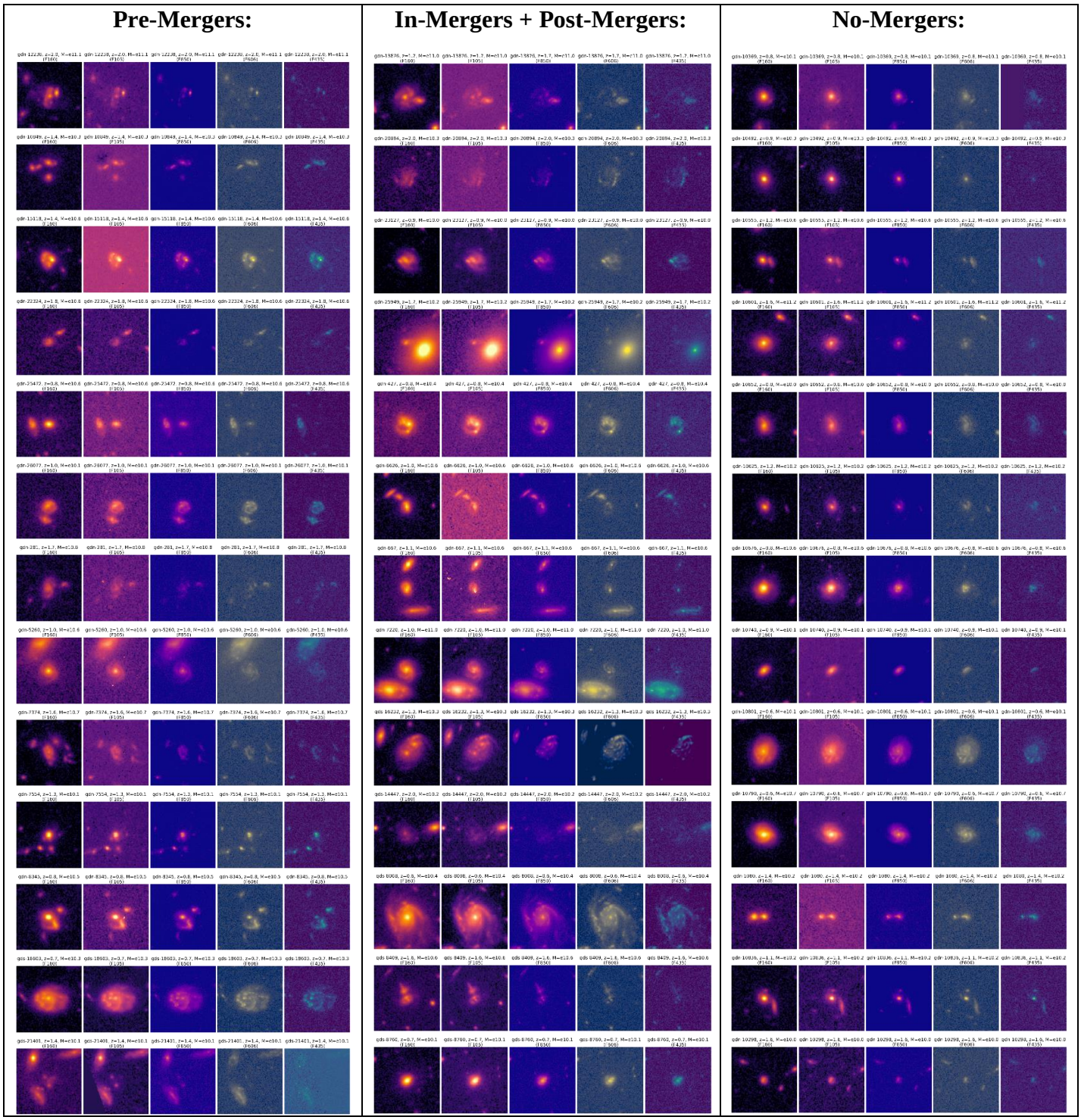


Figure 37: *A sample of real galaxies from CANDELS catalog that were classified as Pre-Mergers (left), In-Mergers + Post-Mergers (center) or No-Mergers (right) by our* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.45+] and* ***preM*** *[-0.3-0] vs* ***(inM + postM)*** *[0-0.3] classifiers. Although the classification might not be perfect, and we cannot really know what's the ground truth, we can still see that the classifications are done well, and the same criteria that the network used on the simulation's mock-images are valid here as well.*

5. Discussion

5.1. Conclusions

In this thesis we developed a tool that detects galaxies that undergo major-merger events using a Machine Learning technology. We trained and tested a series of Convolutional Neural Networks (CNN) on mock-images of galaxies from the Horizon-AGN cosmological simulation that we got from Huertas-Company and we added to them real noise and backgrounds from CANDELS. We then used the networks on real galaxies from 2 CANDELS field: GOODS-S and GOODS-N, to single out galaxies that are during a major merger process, and identified the phase of the merger that they are in. We calculated the merger-fraction and merger-rate as a function of redshift and gave an estimation to the duration of the major-merger event – 0.6 ± 0.1 of the DM halo's dynamical times ("t_{dyn}"). We identified the 3 main phases of the merger and estimated their time-scales:

1. **Pre-Merger** (-0.3 ± 0.05 – $0\ t_{dyn}$) – Starts with the approaching of the progenitors, roughly when their distance is 4 × radius of the main progenitor, and ends at the first closest approach, namely the first pericenter.
2. **In-Merger** (0 – $0.15^{+0.05}_{-0.025}\ t_{dyn}$) – Starts at the first passage and ends at the final coalescence.
3. **Post-Merger** ($0.15^{+0.05}_{-0.025} - 0.3 \pm 0.05\ t_{dyn}$) – Starts at the final coalescence and ends when the galaxy is relaxed and doesn't show serious distorted morphology.

We tested our technique by reproducing another, earlier paper's results (Ferreira et al. 2020): In section 4.1 we trained and tested 2 classifiers with the same duration scales as in the earlier paper: A Mergers [-0.3 – 0.3 Gys] vs No-mergers [0.5+ Gys] classifier, and a Pre-Mergers [-0.3 – 0 Gys] vs (In-Mergers + Post-Mergers) [0 – 0.3 Gys] classifier. The performance of the models over the simulated data were rather good, better than the earlier work. We then applied them to the real CANDELS catalog and reproduced similar merger-rate as a function of redshift that is consistent with the one obtained from different observations and simulations (Ferreira et al. 2020, Duncan et al. 2019). We also got the ratio of Pre-Mergers / In+Post-Mergers = $1.09^{+0.34}_{-0.17}$ (2σ error estimation) among real galaxies from CANDELS, which is more balanced than in Ferreira et al. 2020 (1.42 for $z \leq 2$). Note that we do expect a balanced ratio of ~ 1, because the galaxies in the CANDELS fields are random and we defined the observational time of the 2 classes to be symmetric.

Our ability to reproduce, and even out-perform the results done by other works that used much larger data-sets from a larger simulation, but with a lower resolution – IllustrisTNG300, serves 3 purposes:

1. It is a sanity check of our technique. Both the network architecture, the catalog building and the noise and background sampling.
2. A validation of the results and the legitimacy of both simulations, because we were both able to converge to the same result using different network architectures and different datasets from 2 different simulations.
3. It shows the supremacy of the Horizon-AGN, which has a much smaller number of galaxies but with a higher resolution, and still was able to deliver the same results and allowed us (later) to distinguish between 3 different stages of the merger.

For the comparison above we used absolute time units (Giga years) for the fixed observation time windows. For the rest of the work, unless specified otherwise, we took the time divided by the dark-matter halo's dynamical time of the given redshift of each galaxy: $t \,/\, t_{dyn}$. The reason for it is that t_{dyn} is the natural physical timescale for dynamical processes in the halo, which should apply to mergers.

We showed in section 4.2 that the major-merger event in the simulation also scales as the dynamical time, and then we used it to constrain the end time of the post-merger phase, and by that the duration of the entire major-merger event (subsection 4.2.3), by searching the best time-cut T between early snapshots ($0 < t < T$) and late snapshots ($t > T$) in the "major-mergers" catalog, in terms of the network performance. We have found that $T = 0.3 \pm 0.05\, t_{dyn}$ is the best time-cut, which suggests that around it the galaxies undergo a significant relaxation of some sort, within just $\sim 0.1\, t_{dyn}$. This sudden change might indicate a phase transition (at least observationally). This is a completely objective and unbiased measure of the timescales, which is missing in previous works.

In section 4.3 we also showed that we can separate the middle "In-Merger" phase from the "Post-Merger" phase, with a reasonable measure of success. We gave a time constraint for the end of the middle phase: $0.15^{+0.05}_{-0.025}$ dynamical times after the first passage ($t = 0$). As far as we know, we are the first to do it.

In section 4.4, with the new time constraints and the use of the halos dynamical timescales, we trained and tested another model that detects mergers from no-mergers, a model that separates Pre-Mergers from (In-Mergers + Post-Mergers), applied both of them on real CANDELS images and recalculated the merger-rate. The performance on the simulated images were also very high, and we got the same merger-rate and a similar PreM / (InM + PostM) ratio on real CANDELS as in sections 4.1.1 and 4.1.2. Our merger-rate's magnitude at $z \sim 2$ matches the theory in equation (6) ($0.45\ Gyr^{-1}$) and its power-law within 1σ range is also very close, but not fully match the theoretical value in equation (5) (we got minimum of 2.6 and in equation (5) it's 2.5). Within 2σ range there's a match. Recall that we didn't expect a full match. By measuring it using Deep-Learning and dynamical-timescales, it is probably the most accurate measure done so far.

As far as we know, we are the first to successfully detect major-mergers, their 3 inner phases and objectively estimate the merger-rate using Deep-Learning and a redshift-depended time scales (the halo's dynamical time). Not only that, but we were also able to constrain the duration of the merger event and its inner phase ("In-Merger" – from the first passage to final coalescence). We also point out what morphological features seem to be most important in the classification process (at least for our network), and the fact that it manifests differently in different bands:

- **Major-Mergers** are characterized by tidal features or other distorted morphology, like asymmetries between the shape of the core and the rest of the galaxy (usually a **postM**), or a pair with some distortions (usually the **preM**). The most extreme distortions and possibly 2 cores usually indicate the **inM** phase. They are bright in all the bands.
- **No-Mergers** are "nice", symmetric and they tend to be fainter in the F606 and F435 bands (especially in the outskirts).

* These characterizations aren't always true.

The success with the Horizon-AGN cosmological simulation, which has a higher resolution with the price of a smaller number of galaxies (compared with IllustrisTNG), shows that its "weakness" (the smaller size) isn't impactful at all.

5.2. Caveats

1. Due to a small number of images in the catalogs, especially in high redshifts, we only worked on images of redshifts $z \leq 2$.
2. Throughout this work we filtered-out images of pre-merger galaxies where the 2 progenitors were farther away than $20\ kpc$. The reason is to make sure we capture both progenitors within the image frame. However, It is possible that this fixed distance filter might cause a redshift bias, since it is mostly relevant for massive, low redshift galaxies, where the distances are larger. In appendix 2 we test it and show that the redshift dependency of the results is not affected by it.
3. Throughout this work we assess the uncertainty in the results that comes from variation in the data, the noise/background sampling and the performance of the network over it ("aleatoric uncertainty"). However, another type of uncertainty comes from the very stochastic nature of the training process and the liability of our network architecture and Machine-Learning technology as a whole ("epistemic uncertainty"). For instance, the fact that our network performs better when classifying early (in+post)-merger from late (in+post)-mergers with a time-cut: $t/t_{dyn} = 0.3$, doesn't mean that a different, more sophisticated technology won't find a different result. Indeed, one can see that the error measures and error-bars we have are quite small. Since we can't estimate the epistemic uncertainty we address it differently: In appendix 4 we check what happens when we use different time-cuts and use it to constrain the epistemic uncertainty, at least for this example.
4. Due to a small set of mock-images of the middle "In-Merger" phase (only a $1/8$ of the original data-sets), we estimated its duration ($0.15^{+0.05}_{-0.025}$ dynamical times) using naked "raw" data, without backgrounds or PSF. It eased the task for the network, and because at this point we can probably trust the simulation's sub grid physics (the mergers scales as the halo's dynamical time and the network succeeds on real observations), this time-cut represents the "ground truth" in the simulation (without noise) and perhaps in the real world as well. But, it also means we cannot trust applying this value on a visual classification on real images that do have noise. We first need to repeat it with PSF and backgrounds.

5.3. Future work

There are several interesting things that can be done with Deep Learning:

1. Confirming the duration of the middle "In-Merger" phase with PSF and noise/backgrounds. According to the result, a classifier can be train to see if we can detect this phase in real observations.
2. By using all the multi-phase classifiers: Pre-Mergers, In-Mergers, Post-Mergers and No-Mergers, we can compare the specific-Star-Formation-Rate (sSFR) within each phase in the simulation and real CANDELS. And we can compare it to the estimations in other observations and literature. It can serves as a farther constraint for the epistemic uncertainty of our model, a constraint for the cosmological simulation's sub grid physics and perhaps give us a better understanding of the dependency of sSFR on the merger-phases and other parameters such as the redshift.
3. Another thing is comparing the final post-merger's stellar mass to the sum of the progenitors' mass. How much is lost? If we can compare it to observations it can serves as another constrain of both our technique and the simulation's physics.

Bibliography

Ackermann et al. (2018) : Ackermann, S., Schawinski, K., Zhang, C., Weigel, A. K., & Dennis Turp, M. 2018, Monthly Notices of the Royal Astronomical Society, 479, 415

Almeida et al. 2014 : Almeida, J. S., Elmegreen, B. G., Mu˜noz-Tu˜n´on, C., & Elmegreen, D. M. 2014, Astronomy and Astrophysics Review, 22, 1

Aubert et al., 2004 : Aubert D., Pichon C., Colombi S., 2004, MNRAS, 352, 376

Barnes & Hernquist 1996 : https://iopscience.iop.org/article/10.1086/177957/pdf

Bottrell et al. 2019 : Bottrell, C., Hani, M. H., Teimoorinia, H., et al. 2019, 25, 1. http://arxiv.org/abs/1910.07031

CANDELS – All information is available in the official website: http://arcoiris.ucolick.org/candels/

Cheng et al. 2019 : Cheng, T.-y., Conselice, C. J., Ara, A., & Li, N. 2019

Conselice et al. 2003 : Conselice, C. J. 2003, ApJS, 147, 1 —. 2006, Monthly Notices of the Royal Astronomical Society, 373, 1389

Conselice 2006 : Cheng, T.-y., Conselice, C. J., Ara, A., & Li, N. 2019

Conselice 2014 : Conselice, C. J., Bluck, A. F., Mortlock, A., Palamara, D., & Benson, A. J. 2014, Monthly Notices of the Royal Astronomical Society, 444, 1125

Cook et al. 2000 : Cook, L. T., Zhu, Y., Hall, T. J., & Insana, M. F. 2000, Proceedings of SPIE - The International Society for Optical Engineering, 3982

cox et al. 2008 : https://arxiv.org/pdf/0805.1246.pdf

Dekel & Birnboim 2006 : https://arxiv.org/pdf/astro-ph/0412300.pdf

Dekel et al. 2013 : https://arxiv.org/pdf/1303.3009.pdf

Dubois et al., 2014 : Dubois Y., et al., 2014, MNRAS, 444, 1453

Dubois et al., 2016 : Dubois Y., Peirani S., Pichon C., Devriendt J., Gavazzi R., Welker C., Volonteri M., 2016, MNRAS, 463, 3948

Duncan et al. 2019 : Duncan, K., Conselice, C. J., Mundy, C., et al. 2019, The Astrophysical Journal, 876, 110. http://dx.doi.org/10.3847/1538-4357/ab148a

Ferreira et al. 2020 : https://arxiv.org/pdf/2005.00476.pdf

Goodfellow et al. 2016 : Goodfellow, I., Bengio, Y., & Courville, A. 2016, Deep Learning (MIT Press)

Grogin et al., 2011 : Grogin, N. A., Kocevski, D. D., Faber, S. M., et al. 2011, Astrophysical Journal, Supplement Series, 197, doi:10.1088/0067-0049/197/2/35

Hopkins et al. 2006 : https://arxiv.org/pdf/astro-ph/0603180.pdf

Horizon-AGN – All information is available in the official website : https://www.horizon-simulation.org/about.html

Huertas-Company: https://mhuertascompany.weebly.com/

Huertas-Company et al. 2018 : Huertas-Company, M., Primack, J. R., Dekel, A., et al. 2018, The Astrophysical Journal, 858, 114.

Huertas-Company et al. 2019 : Huertas-Company, M., Rodriguez-Gomez, V., Nelson, D., et al. 2019, 18, 1. http://arxiv.org/abs/1903.07625

IllustrisTNG – All information is available in the official website: : https://www.tng-project.org/dev707/data/landscape/

Kartaltepe et al. (2015) : Kartaltepe, J. S., Mozena, M., Kocevski, D., et al. 2015, The Astrophysical Journal Supplement Series, 221, 11. http://arxiv.org/abs/1401.2455

Koekemoer et al., 2011 : Koekemoer, A. M., Faber, S. M., Ferguson, H. C., et al. 2011, Astrophysical Journal, Supplement Series, 197, doi:10.1088/0067-0049/197/2/36

Komatsu et al., 2011 : Komatsu E., Smith K. M., Dunkley J., et al. 2011, ApJ Sup., 192, 18

Lacey & Cole 1993 :

Lotz et al. 2004 : Lotz, J. M., Primack, J., & Madau, P. 2004, The Astronomical Journal, 128, 163

Lotz et al. 2008 : Lotz, J. M., Jonsson, P., Cox, T. J., & Primack, J. R. 2008, Monthly Notices of the Royal Astronomical Society, 391, 1137

Man et al. 2016 : Man, A. W. S., Zirm, A. W., & Toft, S. 2016, The Astrophysical Journal, 830, 89. http://dx.doi.org/10.3847/0004-637X/830/2/89

Martin et al. 2019 : Martin, G., Kaviraj, S., Hocking, A., Read, S. C., & Geach, J. E. 2019, 1426, 1408. http://arxiv.org/abs/1909.10537

Mo et al. 2010 : Mo, H., van den Bosch, F. C., & White, S. 2010, Galaxy Formation and Evolution https://arxiv.org/pdf/2005.00476.pdf

Neistein & Dekel 2008a,b : Neistein E., Dekel A., 2008, MNRAS, 388, 1792 https://arxiv.org/pdf/0802.0198.pdf

Pearson et al. 2019 : Pearson, W. J., Wang, L., Trayford, J. W., Petrillo, C. E., & van der Tak, F. F. S. 2019, Astronomy & Astrophysics, 626, A49

Reiman & G¨ohre 2019 : Reiman, D. M., & G¨ohre, B. E. 2019, Monthly Notices of the Royal Astronomical Society, 485, 2617

York et al., 2000 :

Appendix

1. Examples for background sampling

In section 2.2.2 we described the process of sampling background stamps from CANDELS fields for the simulated mock-images, with the β parameter condition over the image resulted by a pixel-wise multiplication of the background stamp with the image: $\beta < 4.4$ for a valid stamp and $\beta \geq 4.4$ for an invalid stamp (the condition should be satisfied for all the filters of an image). Figure **38** shows an example of a good stamp with a background galaxy that almost overlaps with the main one, and Figure **39** show an example of a bad stamp for the same image, with a faint background galaxy that overlaps with the main galaxy.

In Figure **40** we show, for comparison, real images from CANDELS next to mock-images from the "isolated" catalog and the "major-mergers" catalog, both with backgrounds.

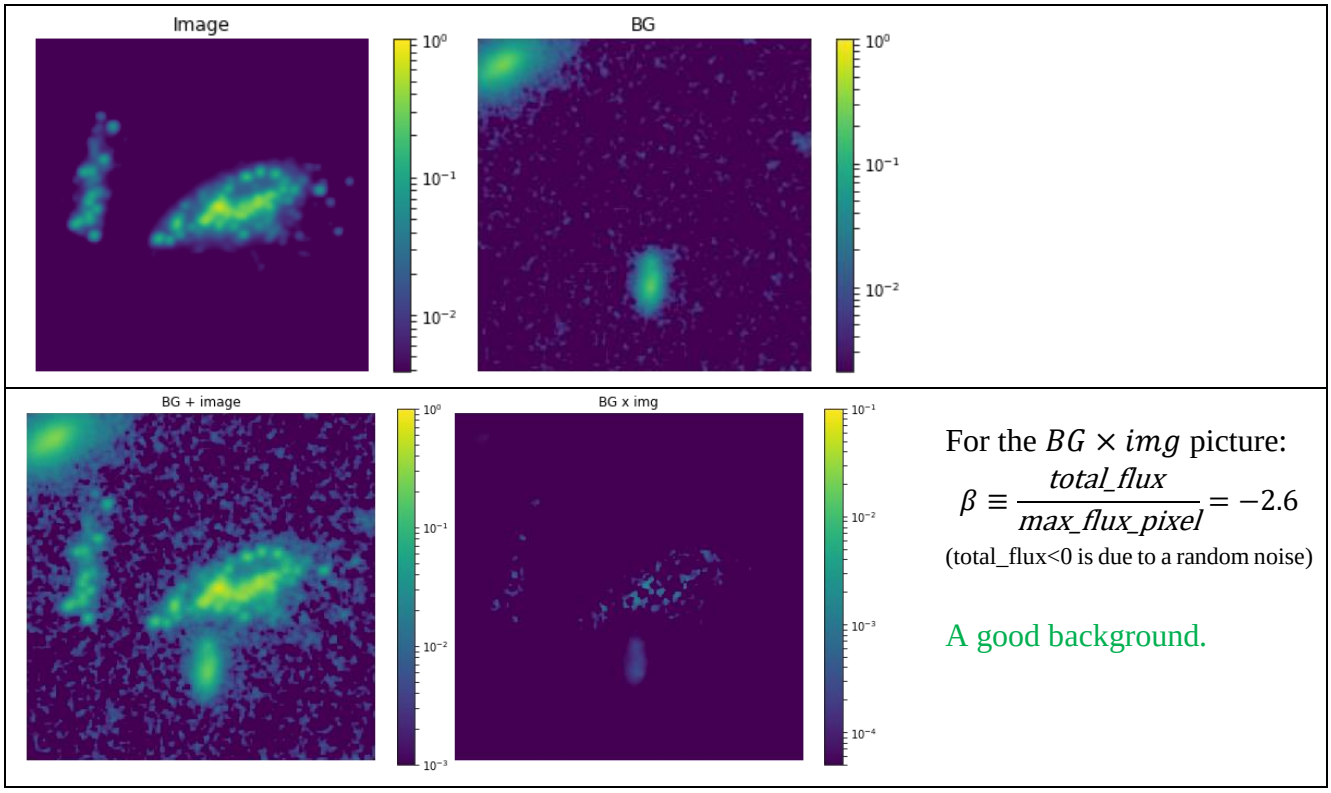


Figure 38: *Example of an approved background stamp. Upper-left image is the original mock-image, upper-right is the background stamp from CANDELS, lower left is the final combination of them and lower right is the pixel-wise multiplication of them. We can see in the multiplied image that the background galaxy is shown, but just barely so $\beta < 4.4$ and it is accepted.*

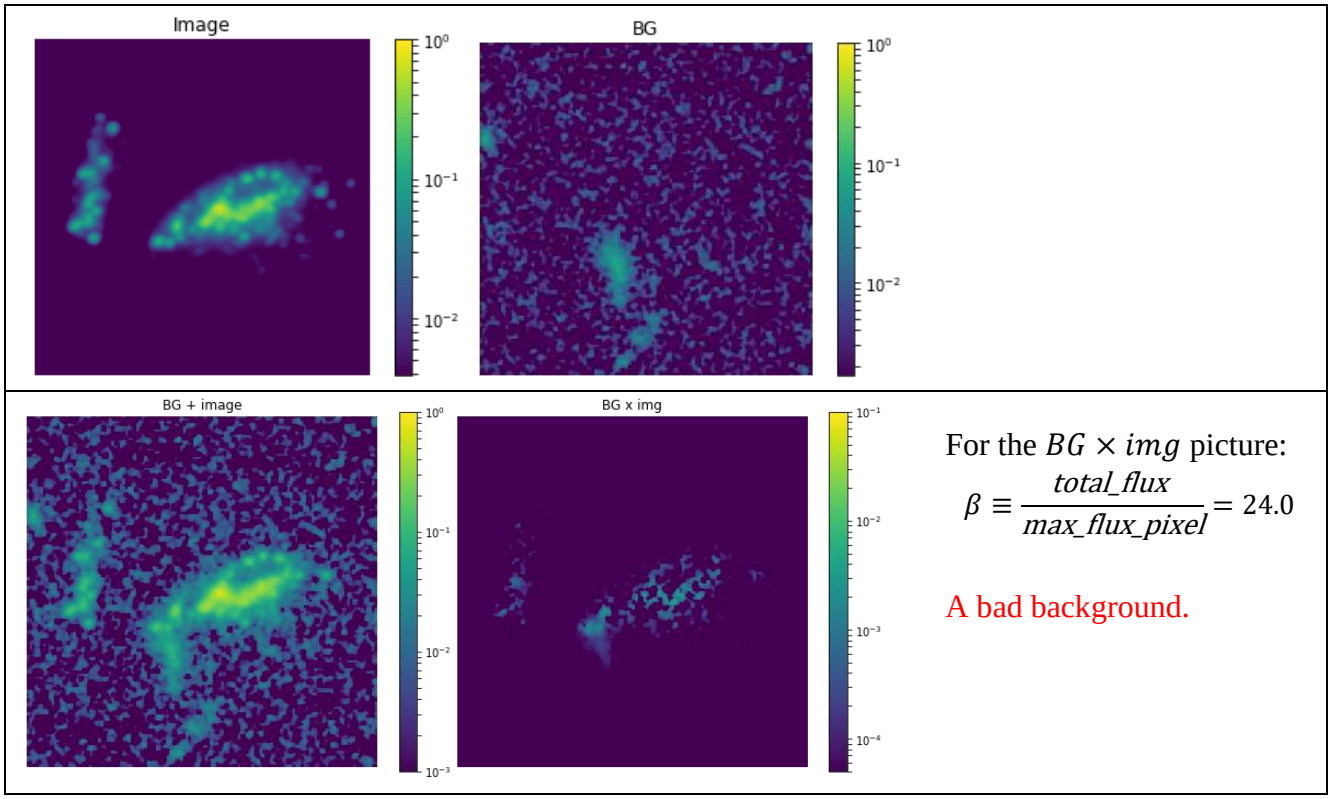


Figure 39: *Example of a rejected background stamp. Upper-left image is the original mock-image, upper-right is the background stamp from CANDELS, lower left is the final combination of them and lower right is the pixel-wise multiplication of them. We can see in the multiplied image that the background galaxy is fainter than the previous example, but closer and overlaps with the image's main galaxy. Therefore* $\beta \geq 4.4$ *and it is rejected.*

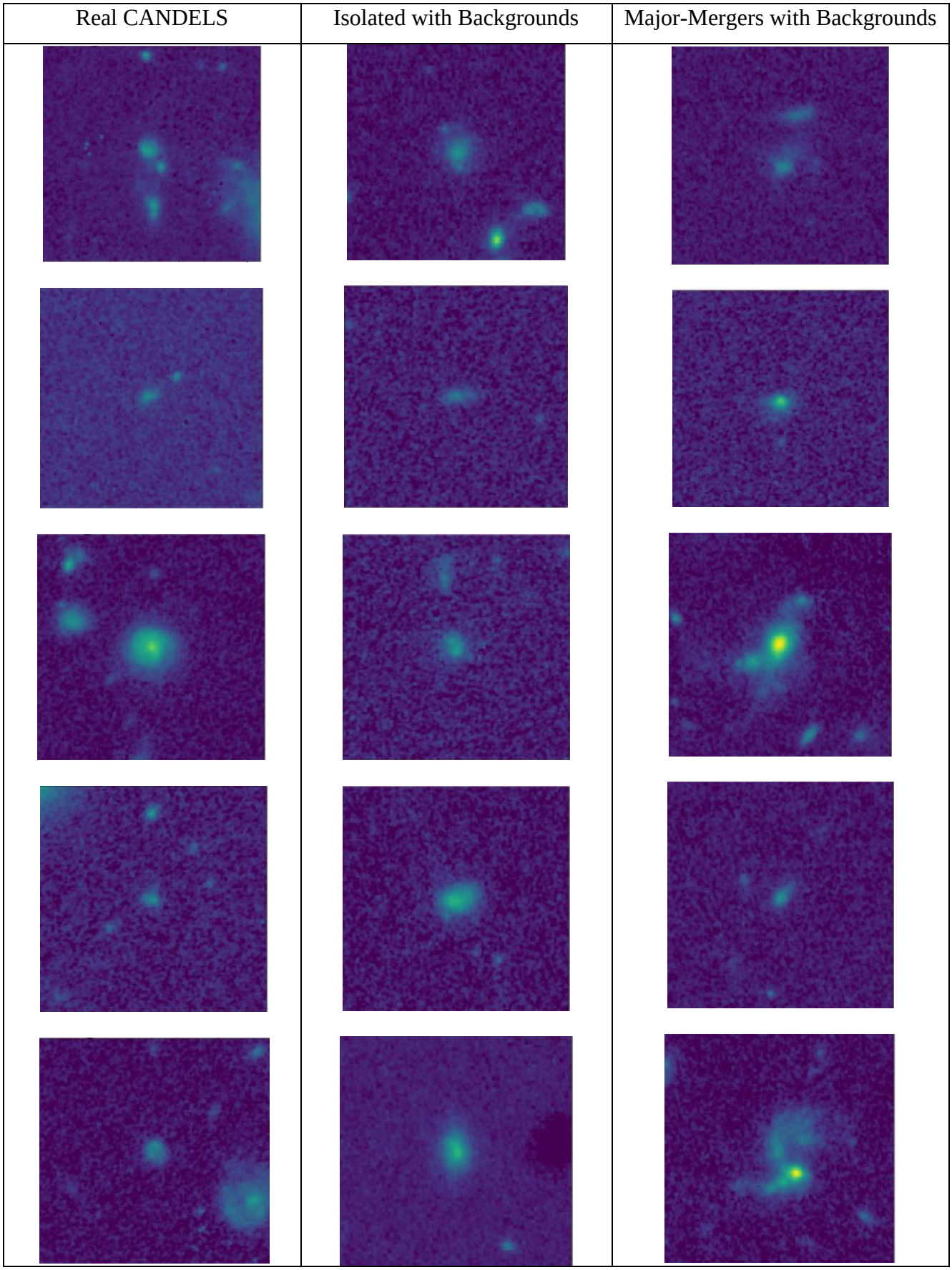


Figure 40: *For comparison – left column is real images from CANDELS, center column is mock-images from the "isolated" catalog and right column is from the "major-mergers" catalog, both with backgrounds.*

2. The impact of a maximal, fixed distance (D ≤ 20 kpc)

Throughout this work we filtered-out images of pre-merger galaxies where the 2 progenitors were farther away than $20\ kpc$. The reason is to make sure we capture both progenitors within the image frame that reaches a physical width of $\sim 45\ kpc$ (on redshift $z = 0.5$). However, one can claim that this fixed distance filter might cause a redshift bias, since it's mostly relevant for massive, low redshift galaxies, where the distances are larger. In order to test it, we trained 2 **merger** vs **no-merger** classifiers, with data-sets that where constructed without this filter:

1. Mergers [-0.3 – 0.3] vs no-mergers [0.5+] in Gys units.
2. Mergers [-0.3 – 0.3] vs no-mergers [0.45+] in dynamical time units.

In the following part we will show the results on the simulation's testing-set, the merger-fraction and merger-rate over real CANDELS images and compare it to with the results of the similar models **with** the fixed maximal distance filter.

Results over the simulation's testing-sets:

Table **15** shows the results of the run over the simulation's testing-set for the case of **merger** [-0.3 – 0.3] vs **no-merger** [0.5+] in Gys units, and Table **16** is the same for the case of **merger** [-0.3 – 0.3] vs **no-merger** [0.45+] in t_{dyn} units. Both with the same metrics as before. We can see that in both cases the accuracy is high (> 0.9), but with a bias of 0.12 (in fixed times case) and 0.14 (in dynamical times case) toward the **merger** class. Figure **41** and Figure **42** show the probability distribution (the confidence) of the predictions for each case. We can see there that the networks are indeed more confident with the **merger** class, especially in the case of fixed real observation times, but there are still high enough confidences in both cases for both classes to allow a small error. The prediction errors and the other error ranges were calculated in the same way as in all the other cases we have ran.

Mergers [-0.3 – 0.3] vs no-mergers [0.5+] in Gys units – Without maximal distance:

Metric	Score
Mean prediction error	0.007
Accuracy	0.908 ± 0.00
Balance-rate	1.120 ± 0.005
Precision	0.964 ± 0.001
Recall	0.848 ± 0.005

Table 15: *The score of the **merger** vs **no-merger** classification in Gys units, without D ≤ 20 kpc filter, over the simulation's testing-set. Accuracy is high but there's a bias of 0.12 toward the **merger** class. The errors represent 1σ prediction-error estimation added to / subtracted from the predictions.*

Mergers [-0.3 – 0.3] vs no-mergers [0.45+] in dynamical-times units – Without maximal distance:

Metric	Score
Mean prediction error	0.007
Accuracy	0.930 ± 0.007
Balance-rate	1.014 ± 0.004
Precision	0.936 ± 0.001
Recall	0.923 ± 0.004

Table 16: *The score of the **merger** vs **no-merger** classification in dynamical time units, without D ≤ 20 kpc filter, over the simulation's testing-set. Accuracy is high but there's a bias of 0.014 toward the **merger** class. The errors represent 1σ prediction-error estimation added to / subtracted from the predictions.*

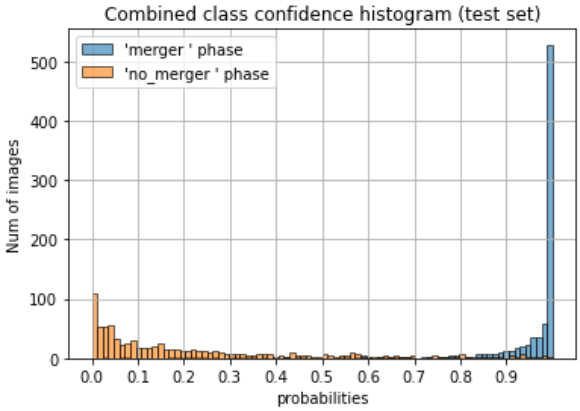


Figure 41: *Probability distribution of the prediction results of the merger vs* ***no-merger*** *classification in Gys units, without $D \leq 20$ kpc filter, over the simulation's testing-set. The classifier is more confident in the* ***merger*** *class than in the* ***no-merger****, which leads to a bias in the results. Both classes are still confident which leads to a very small error.*

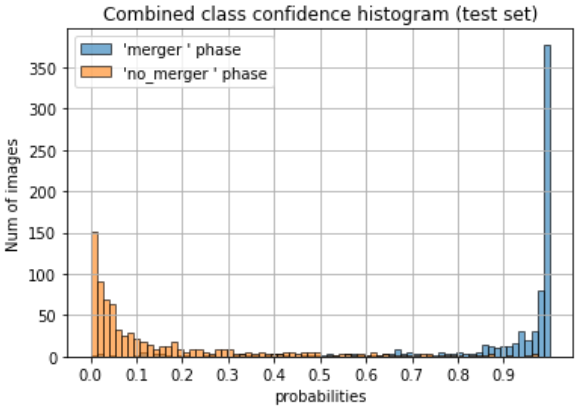


Figure 42: *Probability distribution of the prediction results of the* ***merger*** *vs* ***no-merger*** *classification in dynamical times units, without $D \leq 20$ kpc filter, over the simulated testing-set. There is a bit higher confidence in the* ***merger*** *class than in the* ***no-merger****, which leads to a very small bias in the results. Both classes are very confident → a very small error.*

We then ran each model on real CANDELS galaxies, as before. In each case, we sorted all the images into redshift bins and calculated the merger-fraction and merger-rate, by dividing the merger fraction with the observation time (0.6 Gys for the fixed times case and 0.6 t_{dyn} for the dynamical times case), in each z-bin. Figure **43** and Figure **44** show the results of the absolute-times model and the dynamical-times model (respectively) as functions of redshift in a full logarithmic-scale (labeled as: "Without D"). The merger-fraction are in the left panels and merger-rate in the right panels. For comparison, in each figure we plot in both panels the results we got before **with** the maximal distance filter (labeled as: "With D"). The data of the case **without** the maximal distance filter is coded in blue diamonds, its best fit in a blue dashed line and the fit's 1-STD range is in the blue shade. The data **with** the maximal distance filter is coded in red dots, its best fit in an orange line and the fit's 1-STD range is in the orange shade. We also show in Table **17**) and Table **18** the equations of the merger-fraction and rate of both cases for both models.

For the case of the fixed absolute-time observation window (Figure **43** and Table **17**), we can see, both in the graphs and in the equations, that the merger-rate and merger-fraction are both $2\frac{1}{3}$ times higher without the filter than with it, but the redshift dependency remained within the error range. For the case of the dynamical-time observation window (Figure **44** and Table **18**), the results show a similar behavior, but the rise of the case without the filter is within the error range of the case with the filter, so they're practically match (there's one suspicious point at the lowest redshift, which we can't explain. It could be an error of the model).

We can learn from it that by exposing the network to images of pre-mergers that might show only one progenitor, the network learns to expand the definition of mergers across all redshifts, even if these new images aren't distributed evenly across the redshifts. It is done without updating the observation time which is the reason for the higher merger-rate in the case of the fixed absolute-times models. The conclusion is that by using the maximal-distance filter we don't introduce a redshift bias, but we still preferred to keep it in this project because we are not sure to what extent we can trust the network with images with just 1 progenitor. For example, is the reason of the rise in the merger-rate in the fixed absolute-times model is that the network expanded the observation time for mergers beyond $\pm 0.3\ Gys$? If so, to what value? Is it still symmetric for pre-mergers and post-mergers? Why don't we see it in the dynamical-times model?

Merger-fraction and Merger-rate for
Mergers [-0.3 – 0.3] vs no-mergers [0.5+] in Gys units:

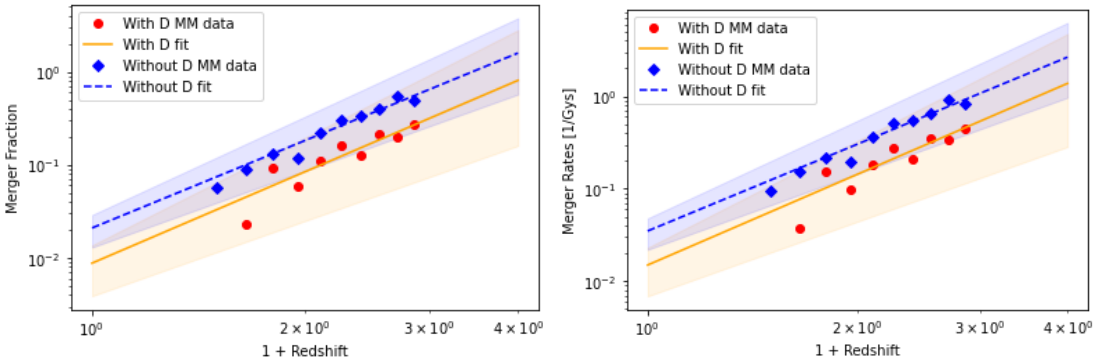


Figure 43: *The merger-fraction (left) and merger-rate (right) as a function of redshift in a full logarithmic scale, for the case of* ***merger*** *vs* ***no-merger*** *classification with Gys units. The results, best fits and fits' 1-STD of the model* ***without*** *max-D filter are the blue diamonds, blue dashed lines and blue shaded areas. The results, best fits and fits' 1-STD of the model* ***with*** *max-D filter are the red dots, orange lines and orange shaded areas. We can see that the redshift dependency remains the same but there's a constant scale increase for the case without max-D filter.*

With max-D:	Without max-D:
$MF = (0.009 \pm 0.005) \cdot (1+z)^{3.26 \pm 0.58}$	$MF = (0.021 \pm 0.008) \cdot (1+z)^{3.12 \pm 0.39}$
$MR = (0.015 \pm 0.008) \cdot (1+z)^{3.26 \pm 0.58} [Gyr^{-1}]$	$MR = (0.035 \pm 0.013) \cdot (1+z)^{3.12 \pm 0.39} [Gyr^{-1}]$

Table 17: *The best fit' equations for the merger-fraction ("MF") and merger-rate ("MR") for the case of* ***merger*** *vs* ***o-merger*** *classification with* ***Gys units****. Left: The model with max-D filter. Right: The model without max-D filter.*

Merger-fraction and Merger-rate for
Mergers [-0.3 – 0.3] vs no-mergers [0.45+] in dynamical-times units:

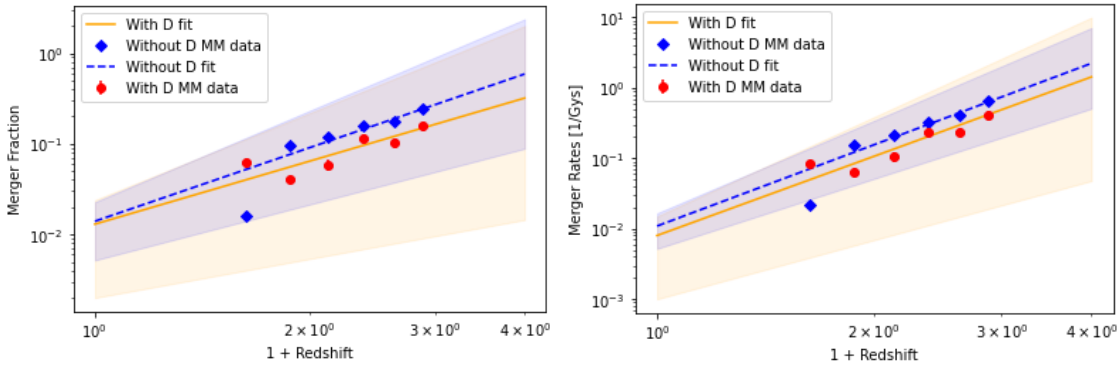


Figure 44: *The merger-fraction (left) and merger-rate (right) as a function of redshift in a full logarithmic scale, for the case of* ***merger*** *VS* ***no-merger*** *classification with dynamical time units. The results, best fits and fits' 1-STD of the model* ***without*** *max-D filter are the blue diamonds, blue dashed lines and blue shaded areas. The results, best fits and fits' 1-STD of the model* ***with*** *max-D filter are the red dots, orange lines and orange shaded areas. We can see that the results are marginally match, with a small constant scale increase for the case without max-D filter, but it's within the error range. The redshift dependency remains the same.*

With max-D:	Without max-D:
$MF = (0.013 \pm 0.011) \cdot (1+z)^{2.31 \pm 0.88}$	$MF = (0.014 \pm 0.009) \cdot (1+z)^{2.70 \pm 0.65}$
$MR = (0.011 \pm 0.009) \cdot (1+z)^{3.55 \pm 0.91} [Gyr^{-1}]$	$MR = (0.011 \pm 0.006) \cdot (1+z)^{3.84 \pm 0.53} [Gyr^{-1}]$

Table 18: *The best fit' equations for the merger-fraction ("MF") and merger-rate ("MR") for the case of* ***merger*** *VS* ***no-merger*** *classification with* ***dynamical time units****. Left: The model with max-D filter. Right: The model without max-D filter.*

3. A possible contradiction between section 4.2.3 and section 4.1.1?

In section 4.2.3 we found that $t/t_{dyn} = 0.3$ makes for the best separation between Post-Mergers and No-Mergers, by giving the most accurate and balanced results, thus defining the end of the major-merger event. The results that we got there seem to be in tension with the results we got from section 4.1.1, where we classified **merger** vs **no-merger** with a fixed duration of [-0.3 – 0.3] Gys (absolute times), with a very high score (in all the metrics). After all, 0.3 Gys means a different fraction of the halo's dynamical time in low redshifts and high redshifts.

Possible explanations:

1. The network in section 4.1.1 (Gys units) was able to learn other features that distinguish the **merger** from the **no-merger** and are invariant in redshift. It contradicts the evolution theory in (1.2) so it is unlikely.
2. For $0.5 \leq z \leq 2$ the time-gap between the **merger** snapshots and the **no-merger** snapshots is wide enough to compensate: In section 4.1.1 for the **no-merger** class we used snapshots of $t \geq 0.5$ Gys since the first passage, and indeed 0.5 Gys > 0.3 $t_{dyn}(z)$ in all the relevant redshifts, so the network should have no problem recognizing the No-Mergers:

$$0.5\ Gys = \begin{cases} \sim 0.4\ t_{dyn}\big|_{z=0.5} \\ 0.5\ t_{dyn}\big|_{z=1} \\ \sim 0.9\ t_{dyn}\big|_{z=2} \end{cases} > 0.3\ t_{dyn}$$

   On the other hand, 0.3 Gys < 0.3 t_{dyn} in all but high redshifts:

$$0.3\ Gys = \begin{cases} \sim 0.2\ t_{dyn}\big|_{z=0.5} < 0.3\ t_{dyn} \\ 0.3\ t_{dyn}\big|_{z=1} = 0.3\ t_{dyn} \\ \sim 0.5\ t_{dyn}\big|_{z=2} > 0.3\ t_{dyn} \end{cases}$$

   So the network in section 4.1.1 (Gys units) should only straggle with part of the galaxies at high redshift ($1^+ < z \leq 2$) that exceed 0.3 t_{dyn}. Fortunately, these "late" snapshots of galaxies from the "major-mergers" catalog at high redshifts are rarer than in low redshifts (we also add galaxies from the "isolated" catalog). And the fact that our **merger** vs **no-merger** classifier in section 4.1.1 got a high score even in high redshifts, is an indication that indeed they are not enough to make a significant bias.

In section 4.4.1 we recalculated the merger-fraction and merger-rate with the new, redshift-depended timescales, and we got merger-rate similar to those we got in section 4.1.1, when using fixed absolute times. It means that our 2nd explanation is plausible and indeed, thanks to the time-gap chosen between **merger** and **no-merger** classes, the images mostly fall, seemingly almost by accident, within the 0.3 halo's dynamical times definition.

Perhaps it shouldn't surprise us that Ferreira et al. 2020 chose this time window (and in section 4.1 we followed them). After all, they had to calibrate their model to match with other works they mentioned like Duncan et al. (2019) and Mundy et al. (2017).

4. Comparing Merger-Rate With different dynamical time cuts

In section 4.2.3 we found out that around $t = 0.3$ dynamical times, merger galaxies undergo a significant visually change, which makes it a good time to mark the end of a "Major-Merger" event. But this result depends heavily on the error-range of our method – the neural network-based machine learning. We have accounted for the uncertainty in the data that comes from background, noise and the confidence of the predictions, but we can't say what's the uncertainty or stability of our network architecture, or machine learning technology as a whole. The fact that machine learning technology is able to classify some cases in a near 100% success (like dogs VS cats), doesn't mean that we can trust this technology for all image processing tasks in the world.

In order to go around this issue, we wish to check what do we get if we do try a different time-cut for the major-mergers, and if the results make sense. If 0.3 dynamical times isn't a "special" time, then by changing the time, to let's say 0.1 dynamical times, we should get a lower merger-fraction but with a proportionally smaller observation time window. Since merger-rate is defined as merger-fraction divided by the observation time at each redshift, we should get the same merger-rate as in the case with the time-cuts $= 0.3\ t_{dyn}$ (for **merger**) and $0.45\ t_{dyn}$ (for **no-merger**. After the gap) shown in section 4.4.1. If however, $0.3\ t_{dyn}$ is a "special" time (at least observationally if not physically), then the network should get confused by labeling visually similar images differently, which leads to a lower score as well as a different merger-rate. A word of caution: Because we still do this test with the same tools – machine learning, then we cannot take it as a perfect test.

Simulation parameters and results:

We trained and tested a Major-Mergers VS No-Mergers classifier with a redshift-depended observation time, on the simulation's mock images with the following parameters:

- Mergers [-0.1 – 0.1] VS no-mergers [0.25+] in dynamical time units.
- Training-set size: 6,336 images.
- Testing-set size: 576 images.

Redshift	Number of images	phase	Number of images
$0.4 \leq z < 1$	2,145	Pre-Merger	1,584
$1 \leq z < 1.5$	2,120	In+Post -Merger	1,584
$1.5 \leq z < 2$	2,071	No-Merger	3,168

Table 19: *The distribution of the training-set's mock images from the simulation after they were balanced by redshift bins (**left**) and phase (**right**). The case is of **merger** [-0.1 – 0.1] VS **o-merger** [0.25+] in dynamical time units. Notice that the number of **no-merger** images equals to the number of **preM** + **inM** + **postM** and the redshift bins are also balanced with up to 3% difference.*

Notice that we took a "gap" of $0.15\ t_{dyn}$ between the **merger** class and the **no-merger** class. It required for a fair comparison. Also notice that the **no-merger** class now includes images of $t/t_{dyn} < 0.3$. Table **19** shows the number of images in the training-set, and their distribution over redshifts and phases. We can see that they are almost perfectly balanced with up to 3% difference between the number of images within different redshift bins. After the training we evaluate the model over a testing set like

we did in the rest of the work, resulting in a prediction/probability vector that we cast to either **merger** or **no-merger** classes, and an 1σ error estimation for the predictions.

Table **20** shows the results according to the 2 interchangeable metrics: "Accuracy" and "Balance/difference-rate", "Precision" and "Recall". We can see that they're highly balanced (a bias of 0.011), but the rest of the values are around ~0.8 score, which is lower than in the case of time-cuts = 0.3, 0.45 t_{dyn} from section 4.4.1 (~0.9). Figure **45** shows the confusion matrix with the same coding as before. We can see that the network straggles and classifies wrongly 20% of the images, which is already an alarming result because with different time-cuts (0.3 t_{dyn}) we got better performance.

Metric	Score
Mean prediction error	0.0078
Accuracy	0.801 ± 0.017
Balance-rate	1.011 ± 0.014
Precision	0.804 ± 0.003
Recall	0.795 ± 0.014

Table 20: *The score of the **merger** [-0.1 – 0.1] VS **no-merger** [0.45+] classification in dynamical time units, over the simulation's testing-set. The Metrics' meaning are explained in section 3.3 and the errors represent 1σ prediction-error estimation added to / subtracted from the predictions.*

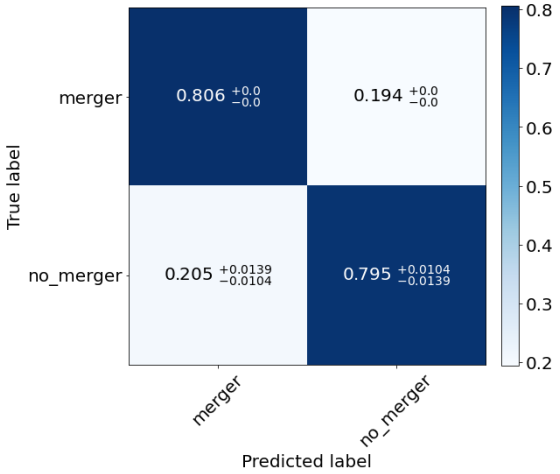


Figure 45: *Normalized confusion matrix for the **merger** [-0.1 – 0.1] VS **no-merger** [0.25+] classification results in dynamical time units, over the simulation's testing-set. Each row represents the true labels of the relevant class, and each column represents the predicted label. The results are highly balanced with a small bias (0.011) toward the **merger** class. The small errors mean a high confidence for the predictions.*

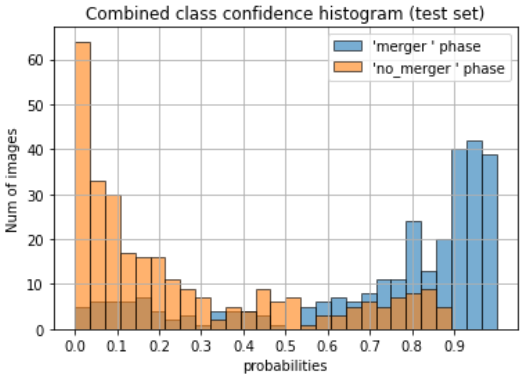


Figure 46: *Probability distribution of the prediction results of the **merger** [-0.1 – 0.1] VS **no-merger** [0.25+] classification in dynamical time units, over the simulation's testing-set. The classifier is more confident in the **merger** class than in the **no-merger** one, but the confusion seems on the same scale.*

Real CANDELS results:

We then used the model on real CANDELS galaxies, sorted the classified images into several redshift bins and calculated the merger-rate (by calculating the merger-fraction and then divide it by the observation time of 0.2 dynamical times). We estimated the uncertainty in the merger-rate by adding/subtracting the 1σ prediction-errors that we got in the simulation, to/from the prediction results of the real CANDELS, and then casting each prediction to either **merger** or **no-merger**.

In Figure **47** we plot the merger-rate as a function of redshift (green squares) with its best fit (dotted, green line) and the fit's 1-STD (green shade). For comparison, we also add the merger-rate (red dots),

best fit (orange line) and the fit's 1-STD (orange shade) of the case of **merger** [-0.3 – 0.3] VS **no-merger** [0.45+] (dynamical time units). Left panel is in half-logarithmic scale and right panel is in full logarithmic scale. We also show the formulas of both fits above the graphs.

We can see that the redshift dependencies of both models match, but there's a constant scale factor of ~2.7 for the [-0.1 – 0.1] model. This scale difference puts the models just barely within one another's error range. The seemingly failure to produce the same merger-rate is an indication that at least one of the time-cuts chosen confuses the network, making it over/underestimate the merger-fraction within the same observation-time. The fact that the redshift dependency doesn't change shows the strength of working with dynamical times, as the confusion is of the same proportion in all redshifts. As expected. Since the merger-rate that we got in the case of [-0.3 – 0.3] dynamical time matches other works, and also repeatedly gives higher scores over the mock testing-set, this is the value we are going to trust. It also serves as a constraint for our uncertainty of the results in section 4.2.3.

Mergers [-0.1 – 0.1] vs No-Mergers [0.25+]: $MR = (0.029 \pm 0.018) \cdot (1+z)^{3.84 \pm 0.63} [Gyr^{-1}]$ (22)

Mergers [-0.3 – 0.3] vs No-Mergers [0.45+]: $MR = (0.011 \pm 0.009) \cdot (1+z)^{3.55 \pm 0.91} [Gyr^{-1}]$ (23)

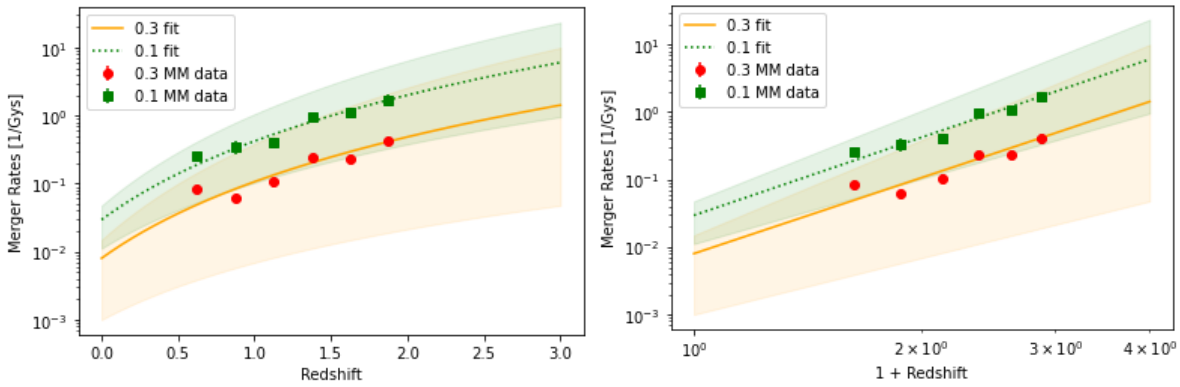


Figure 47: *Left: Merger-Rate vs redshift in a half-log scale. Right: Merger-Rate VS (1+redshift) in a log-log scale. The case is of* ***merger*** *[-0.1 – 0.1] VS* ***no-merger*** *[0.25+] in dynamical time units, over real CANDELS images (green squares). The dotted green lines are the best fit and the green shaded areas are the 1-σ error-range of the fits. We add the results of the case* ***merger*** *[-0.3 – 0.3] vs* ***no-merger*** *[0.45+] in dynamical times for comparison (red dots and orange line and shaded area). We can see that there's a factor of ~3 between the cases which means that one of the merger's time-cuts (probably 0.1* t_{dyn}*) is problematic.*