introduce a new method that provides significantly better results on both clean and noisy maps.

Recent advances in the field of machine learning (ML) have led to new tools in weak-lensing data analysis to break the well-known degeneracy between standard parameters \( \sigma_8 \) and \( \Omega_m \). For example, a Convolutional Neural Network (CNN) trained on (noisy) convergence maps can discriminate models along this degeneracy better than skewness and kurtosis [18]. CNNs have also obtained tighter constraints on the \( \sigma_8-\Omega_m \) confidence contour than traditional Gaussian (e.g. power spectrum) and non-Gaussian (e.g. peak statistics) observables [19, 20]. An interpretation of the filters learned by such networks points to the steepness (as opposed to the height) of local peaks as carrying the most salient cosmological information [21]. Beyond the weak-lensing domain, ML algorithms have also been applied to investigate large-scale structure formation [22, 23] and the connection between baryonic and dark matter [24–26].

In light of P18 results, we aim to determine whether ML can discriminate better than peak counts between \( \Lambda \)CDM and MG models with massive neutrinos. We study weak-lensing maps as in P18 from a subset of the DUSTGRAIN-pathfinder simulations [27], which represent viable locations in the \( f_{R0} - M_\nu \) parameter space but are difficult to distinguish from \( \Lambda \)CDM via only second-order statistics. We consider one \( \Lambda \)CDM model without neutrinos and three \( f(R) \) modified gravity models with different neutrino mass sums \( M_\nu := \sum m_\nu \in \{0, 0.1, 0.15\} \) eV, where we fix the parameters \( n = 1 \) and \( f_{R0} = -10^{-5} \) in the Hu–Sawicki formulation [28]. These simulations have been used as well to forecast cosmological constraints using the dark matter halo mass function [29] and to compare more general map descriptors and ML classification schemes on the full set of nine available models [30].

Given the simulation parameters, we obtained 256 pseudo-independent convergence maps for each model us-
ing the MAPSim pipeline [31]. As the past light cones were produced by randomly reorienting the same particle fields, different lines of sight are not strictly independent; however, we do not expect this approximation to impact our analysis. Each map covers a 25 deg² square area of sky sampled at 2048 × 2048 pixels. Six maps were computed for each line of sight assuming source galaxy redshift planes at \( z_s \in \{0.5, 1.0, 1.5, 2.0, 3.0, 4.0\} \). We consider only the four redshifts up to \( z_s = 2.0 \) in order to compare with previous results [P18].

Using such large maps as direct inputs to a CNN represents a computational challenge given limited resources. We propose to alleviate this issue by reducing the dimensionality of our data before training, that is, by condensing each map’s information content. Our approach uses the probability distribution function (PDF) of map coefficients after a multi-scale wavelet transform.

**Wavelet PDF representation.**—Wavelet transforms have found numerous applications in astronomical image processing. In particular, the starlet (isotropic undecimated wavelet) transform provides a useful representation space for weak-lensing convergence maps [32–39]. This transform naturally facilitates a multi-scale analysis: an initial \( N \times N \) map is decomposed into \( j_{\text{max}} \) wavelet coefficient maps labeled as \( w_j, j \in \{1, ..., j_{\text{max}}\} \), plus a final coarse-scale map. Each \( N \times N \) \( w_j \) is equivalent to an aperture mass map [34] filtered at a scale of \( 2^j \) pixels.

Starting from a convergence map, we derive its condensed PDF representation as follows: (1) compute the starlet transform with \( j_{\text{max}} = 5 \), corresponding to a minimum (maximum) aperture smoothing of 0.293 (4.69) arcmin; (2) divide each \( w_j \) map by the standard deviation of all 1024 maps (256 per model) at the same redshift and scale; (3) bin the resulting pixels by value into arrays with 100 steps between \(-1.0\) and 1.0 per scale; (4) stack the arrays from each scale to produce a \( 5 \times 100 \) matrix; (5) stack the matrices from each redshift to produce a \( 4 \times 5 \times 100 \) datacube. The normalization in step (2) serves to standardize the data so that values across different redshifts can be meaningfully compared by the network. Although the natural domain for each PDF is scale-dependent, we have chosen bin values such that the full information at each scale is included.

The resulting PDF representation of an example ΛCDM map is shown in Fig. 1, separated along the redshift axis. The matrices amount to the combined PDFs across all wavelet scales of interest and reduce the input data dimensionality by a factor of \( \sim 8400 \). Moreover, they are concatenated in a way that should allow the CNN to learn trends reflecting the variation across scales as well as the evolution with redshift. We have also explored an alternative representation based on wavelet peak counts, but since it gives similar final results, we only present the PDF representation here.

**CNN architecture.**—We build a CNN to perform cosmological model classification based on the \( 4 \times 5 \times 100 \) inputs computed from each convergence map. The architecture is detailed in Table I. All activation functions are taken to be rectified linear units [i.e. ReLU, 40]. The first two convolution layers allow the network to extract features related to local variations in the datacube. Each convolution layer has eight output filters to store the different properties found, and the \( 2 \times 3 \times 10 \) size is to explore correlations in the PDFs both within a scale as well as across neighboring scales and redshifts.

A max-pooling operation follows the convolutions and reduces the number of parameters with a mask of size \( 1 \times 1 \times 5 \). This condenses local features into their most relevant pixels while preserving the spatial structure of the data. Another pair of convolution and max-pooling layers constitute the fourth and fifth layers. The same kernel as above is used for the convolution, but the max-pooling layer uses a smaller mask \( 1 \times 1 \times 2 \) to reduce the information now by a factor of two.

Before flattening the result into a one-dimensional array of size 1600, we perform a 30% dropout, which helps prevent over-fitting of the training data. The last three

---

**TABLE I. Architecture of our CNN.**

| Layer type     | Output shape | # params |
|----------------|--------------|----------|
| Input layer    | 1 × 4 × 5 × 100 | 0        |
| Conv 3D [2 × 3 × 10] | 8 × 4 × 5 × 100 | 448      |
| Conv 3D [2 × 3 × 10] | 8 × 4 × 5 × 100 | 3848     |
| Max pooling [1 × 1 × 5] | 8 × 4 × 5 × 20 | 0        |
| Conv 3D [2 × 3 × 10] | 8 × 4 × 5 × 20 | 3848     |
| Max pooling [1 × 1 × 2] | 8 × 4 × 5 × 10 | 0        |
| Dropout [0.3]  | 8 × 4 × 5 × 10 | 0        |
| Flatten        | 1600         | 0        |
| Fully connected | 32           | 51232    |
| Fully connected | 16           | 528      |
| Fully connected | 4            | 68       |
layers are fully connected and end with a softmax function for the actual classification. The final output layer has four nodes corresponding to the four cosmological models that label our data.

Generating noisy data.—Convergence maps derived from real data (i.e. galaxy shape catalogs) are subject to many sources of noise, such as intrinsic galaxy shapes, masking and border effects during shear inversion, instrumental noise, and shape measurement errors. To assess the robustness of our network to noise, we generate noisy versions of each convergence map. Following P18 (cf. the Appendix), we add a random Gaussian noise component with standard deviation $\sigma = 0.7$ and zero mean—a conservative (pessimistic) scenario. For comparison, we also explore a more optimistic scenario with $\sigma = 0.35$, which could arise, for example, from a deeper galaxy survey with a larger number density of objects.

Noise is applied directly to the convergence maps before performing the wavelet transform in the dimensionality reduction scheme. In the noisy analysis, the PDF representations are computed the same way as before, meaning we do not include any additional de-noising step. The result for each map is a datacube of the same dimension as depicted in Fig. 1 but now blurred.

Training strategy.—Our cosmological model discrimination problem in the neural network framework becomes a probabilistic classification problem; we thus use the categorical cross-entropy as the CNN’s loss function to minimize. To train the network, we use the Adam [41] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ (default Keras1 parameters), and an initial learning rate of $\eta_0 = 0.001$. We add a decay parameter $\gamma = 0.001$ to mildly decrease the value of the learning rate throughout epochs according to $\eta(t) = \eta_0(1 + \gamma t)^{-1}$, where $t$ identifies the update step. One step corresponds to one update through a batch of chosen size, which we take to be 200.

We train the CNN on correctly labeled input data for 1000 epochs using 75% of each model’s full dataset, randomly selected. The remaining 25% is reserved for testing the network. Consequently, it takes four update steps to complete one epoch and 4000 steps to finish training. The learning rate is five times smaller at the end of training than at the beginning, which allows for smoother convergence towards the end. We record the loss function and validation accuracy after each epoch (see below), computed on the training and test data, respectively, to monitor the training progress.

Network performance measures.—Training can be monitored by the computation of several metrics. One is the loss (or cost) function, which measures the discrepancy between the targeted output and the prediction of the network. As mentioned above, the standard loss function for classification problems is based on the categorical cross-entropy. Given two discrete probability distributions $p$ and $q$ over the same set of events $\Omega$, cross-entropy $l$ is computed as

$$l(p, q) = -\sum_{\omega \in \Omega} p(\omega) \log q(\omega),$$

where $q$ denotes the predicted probability and $p$ the desired output. For our purposes, the set of events corresponds to the four model labels. After each epoch, the loss function is the sum of the cross-entropies over the training data:

$$\mathcal{L} = \sum_{i=1}^{n} l(p^{(i)}, q^{(i)}),$$

where $n = 768$ is the number of training datacubes.

The network’s learning procedure consists in changing its parameter values (i.e. weights) via stochastic gradi-

---

1. https://keras.io
ent descent and back-propagation in order to make predictions that better match the desired output (i.e. that minimize the loss function). The smaller the loss function, the better the network has learned the relationship between the training data and their corresponding labels. By tracking the loss function over epochs, as shown in Fig. 2 for the three noise levels, one can determine whether the network has trained sufficiently.

Another measure of a network’s performance is its validation accuracy, or the ratio of correct predictions to the total number of test observations. Validation accuracy is an indicator of the predictive power of the network and measures its ability to generalize to new data. Plotted in Fig. 3 is the evolution of the validation accuracy over 1000 epochs computed on the test data. The ratio of correct predictions decreases from about 92% to 48% as the noise standard deviation increases from 0 to 0.7.

Evaluation strategy.—The final assessment of the CNN is obtained with a confusion matrix, computed on the test data, which characterizes the ability of the network to classify each cosmological model. A confusion matrix is a square array whose rows correspond to the true model labels and whose columns represent the predictions. The values in a given row add up to unity and indicate the probability of the network to classify a map from the input model across each of the available output labels. The closer a confusion matrix is to the identity, the better is the classification power of our network.

Our final reported numbers represent the average confusion matrix calculated over the matrices obtained from 100 iterations of the complete training process, where each iteration used a different random selection of training and test maps. An iteration of 1000 epochs required about 13 minutes, so the entire training process took approximately 20 hours.

Results.—We present the final confusion matrices of our CNN in Fig. 4 along with a comparison to P18 results. In the noise-free case (upper plot), the CNN is able to discriminate between all four models with at least 83% accuracy, notably reaching 100% for the two most degenerate models using linear, Gaussian, and non-Gaussian statistics [P18]: ΛCDM and $f(R)$ with $M_\nu = 0.15$ eV. Classification accuracy is also high among the different $f(R)$ models, with the largest confusion (13%) arising between the $M_\nu = 0$ eV and $M_\nu = 0.1$ eV cases.

With noise added (middle and lower plots), confusion increases between many pairs of models, as indicated by the larger off-diagonal values. When $\sigma_{\text{noise}} = 0.35$, ΛCDM becomes somewhat confused with $f_0(R)$ with $M_\nu = 0.15$ eV, although less so with the other $f_0(R)$ models. The confusion between the 0 eV and 0.1 eV $f_0(R)$ cases increases as well, and these trends are further amplified at the highest noise level ($\sigma_{\text{noise}} = 0.7$).

In P18, the ability of different non-Gaussian statistics to distinguish between cosmological models was measured by the so-called discrimination efficiency $\mathcal{E}$. Reported as a percentage, 0% being indistinguishable and 100% being fully distinguishable, the discrimination efficiency was calculated via the False Discovery Rate (FDR) technique of Benjamini and Hochberg [42]. In this scheme, given a statistic, wavelet scale, and source redshift, one computes $\mathcal{E}$ for one model (prediction) taking another model as reference (truth), and the result need not be identical if the models are reversed.

We compare the CNN results of this work with peak counts from P18 by translating the latter into confusion values, shown in the lower right of each cell of Fig. 4, as follows. Given a true reference model, we compute $\mathcal{E}$ for peak counts at $z_\ast = 2.0$ across the same five wavelet scales used by the CNN for each test model. We have fixed $z_\ast = 2.0$ for simplicity and because higher source redshifts tended to yield better results, as the convergence maps incorporate more information about cosmological structure useful in identifying them. We record

| $\sigma_{\text{noise}} = 0$ | $f_0(R)$ | $f_1(R)$ | $f_2(R)$ | $f_3(R)$ |
|---------------------------|----------|----------|----------|----------|
| $\Lambda$CDM              | $M_\nu = 0$ eV | $M_\nu = 0.1$ eV | $M_\nu = 0.15$ eV |
| $f_0(R)$                   | 1.00     | 0.00     | 0.00     | 0.00     |
| $f_1(R)$                   | 0.00     | 0.88     | 0.12     | 0.00     |
| $f_2(R)$                   | 0.00     | 0.13     | 0.83     | 0.04     |
| $f_3(R)$                   | 0.00     | 0.00     | 0.09     | 0.96     |

FIG. 4. Confusion matrices from the trained CNN on test data. Without noise, the CNN is able to discriminate all four models from each other with at least 83% accuracy, including among the three $f(R)$ models with different neutrino masses. The rate of successful predictions decreases with increasing noise, but even for the pessimistic case ($\sigma_{\text{noise}} = 0.7$) the CNN retains non-negligible discrimination power. In the lower right of each cell is the best case value using the thresholded peak count statistics of P18 for $z_\ast = 2.0$. The CNN performs significantly better in every case (i.e. is closer to the identity matrix), and the predictive power of peaks approaches zero with increasing noise.
in the confusion matrix the highest value attained at any wavelet scale and then normalize each row to sum to one. The values therefore constitute the best case achievable by peak count statistics using the previous technique.

The CNN confusion matrices are closer to the identity in each noise case compared to peaks. Without noise, both the CNN and peaks are able to fully distinguish the three MG models from ΛCDM. Indeed, the result for peaks was one of the main conclusions of P18. However, among the three $f_5(R)$ models, the CNN is more sensitive to the value of $M_\odot$, as the correct model is identified with peaks only 44%–66% of the time. Furthermore, whereas the CNN retains non-negligible predictive power with rising noise, the prediction rate of peaks becomes consistent with random guessing (i.e. 25%).

Conclusions.—Weak-lensing convergence maps carry significant information that, if properly extracted and leveraged, can identify their underlying cosmology. To this end, we designed a CNN to distinguish among degenerate cosmological scenarios using the similar convergence maps they produce. We tested our method on four simulated models occupying positions in the joint parameter space of $f(R)$ models and massive neutrinos, designed to be difficult to discriminate based on their two-point and higher-order statistics. Breaking such degeneracies in observables with ΛCDM represents an important step toward better understanding the nature of cosmic acceleration.

To alleviate the computational cost of training on full maps, we reduced the dimensionality of the problem using a novel data representation based on multi-scale wavelet PDF coefficients. Compared to the state-of-the-art using peak counts [P18], our CNN performed significantly better in terms of the confusion matrix between models for three different noise levels. While peak counts significantly better in terms of the confusion matrix between art using peak counts [P18], our CNN performed significantly better in terms of the confusion matrix between art using peak counts [P18], our CNN performed significantly better in terms of the confusion matrix between

The Python code with instructions for reproducing our results is available at http://www.cosmostat.org/software/mgcnn.

Acknowledgments.—AP acknowledges support from an Enhanced Eurotalents Fellowship, a Marie Skłodowska-Curie Actions Programme co-funded by the European Commission and Commissariat à l’Énergie Atomique et aux Énergies Alternatives (CEA). JM has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 664931. CG and MB acknowledge support from the Italian Ministry for Education, University and Research (MIUR) through the SIR individual grant SIMCODE (project number RBSI14P4IH). JM, CG, and MM acknowledge support from the Italian Ministry of Foreign Affairs and International Cooperation (MAECI), Directorate General for Country Promotion (project “Crack the lens”). We also acknowledge the support from the grant MIUR PRIN 2015 “Cosmology and Fundamental Physics: illuminating the Dark Universe with Euclid”; and the financial contribution from the agreement ASI n.I/023/12/0 “Attivit`a relative alla fase B2/C per la missione Euclid”. The DUSTGRAIN-pathfinder simulations discussed in this work have been performed and analyzed on the Marconi supercomputing machine at Cineca thanks to the PRACE project SIMCODE (grant No. 2016153604) and on the computing facilities of the Computational Center for Particle and Astrophysics (C2PAP) and of the Leibniz Supercomputer Center (LRZ) under the project ID pr94i.

* austin.peel@cea.fr

[1] Planck Collaboration, P. A. R. Ade, N. Aghanim, M. Arnaud, M. Ashdown, J. Aumont, C. Baccigalupi, A. J. Banday, R. B. Barreiro, N. Bartolo, E. Battaner, R. Battye, K. Benabed, A. Benoit, A. Benoit-Lévy, J. P. Bernard, and others, A&A 594, A14 (2016), arXiv:1502.01590.

[2] Planck Collaboration, N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, M. Ballardini, A. J. Banday, R. B. Barreiro, N. Bartolo, S. Basak, R. Battye, K. Benabed, J.-P. Bernard, M. Bersanelli, P. Bielewicz, and others, ArXiv e-prints (2018), arXiv:1807.06209.

[3] H. Motohashi, A. A. Starobinsky, and J. Yokoyama, Physical Review Letters 110, 121302 (2013), arXiv:1203.6828 [astro-ph.CO].

[4] J.-h. He, Phys. Rev. D 88, 103523 (2013), arXiv:1307.4876 [astro-ph.CO].

[5] M. Baldi, F. Villaescusa-Navarro, M. Viel, E. Puchwein, V. Springel, and L. Moscardini, MNRAS 440, 75 (2014), arXiv:1311.2588.

[6] B. S. Wright, H. A. Winther, and K. Koyama, J. Cosmology Astropart. Phys. 10, 054 (2014), arXiv:1705.08165.

[7] S. Tsujikawa and T. Tatekawa, Physics Letters B 665, 325 (2008), arXiv:0804.4343.

[8] F. Schmidt, Phys. Rev. D 78, 043002 (2008), arXiv:0805.4812.

[9] E. Beynon, D. J. Bacon, and K. Koyama, MNRAS 403, 353 (2010), arXiv:0910.1480.

[10] S. F. Daniel, E. V. Linder, T. L. Smith, R. R. Caldwell, A. Cooray, A. Leauthaud, and L. Lombrosi, Phys. Rev. D 81, 123508 (2010), arXiv:1002.1962.

[11] I. Tereno, E. Sembolini, and T. Schrabback, A&A 530, A68 (2011), arXiv:1012.5854 [astro-ph.CO].

[12] S. Camera, A. Diaferio, and V. F. Cardone, J. Cosmology Astropart. Phys. 7, 016 (2011), arXiv:1104.2740.

[13] C. Heymans, E. Grocutt, A. Heavens, M. Kilbinger, T. D. Kitching, F. Simpson, J. Benjamin, T. Erben, H. Hildebrandt, H. Hoekstra, Y. Mellier, L. Miller, L. Van Waerbeke, M. L. Brown, J. Coupon, and others, MNRAS
[14] C. D. Leonard, T. Baker, and P. G. Ferreira, Phys. Rev. D 91, 083504 (2015), arXiv:1501.03509.
[15] H. Hildebrandt, M. Viola, C. Heymans, S. Joudaki, K. Kuijken, C. Blake, T. Erben, B. Joachimi, D. Klaes, L. Miller, C. B. Morrison, R. Nakajima, G. Verdoes Kleijn, A. Amon, A. Choi, and others, MNRAS 465, 1454 (2017), arXiv:1606.05338.
[16] DES Collaboration, T. M. C. Abbott, F. B. Abdalla, A. Alarcon, J. Aleksić, S. Allam, S. Allen, A. Amara, J. Annis, J. Asorey, S. Avila, D. Bacon, E. Balbinot, M. Banerji, N. Banik, W. Barkhouse, and others, ArXiv e-prints (2017), arXiv:1708.01530.
[17] A. Peel, V. Pettorino, C. Giocoli, J.-L. Starck, and M. Baldi, ArXiv e-prints (2018), arXiv:1805.05014.
[18] J. Schmelzle, A. Lucchi, T. Kacprzak, A. Amara, R. Sgier, A. Réfrégier, and T. Hofmann, ArXiv e-prints (2017), arXiv:1707.05167.
[19] A. Gupta, J. M. Z. Matilla, D. Hsu, and Z. Haiman, Phys. Rev. D 97, 103515 (2018), arXiv:1802.01212.
[20] J. Fluri, T. Kacprzak, A. Lucchi, A. Réfrégier, A. Amara, and T. Hofmann, ArXiv e-prints (2018), arXiv:1807.08732.
[21] D. Ribli, B. Armin Pataki, and I. Csabai, ArXiv e-prints (2018), arXiv:1806.05995.
[22] L. Lucie-Smith, H. V. Peiris, A. Pontzen, and M. Lochner, MNRAS 479, 3405 (2018), arXiv:1802.04271.
[23] P. Berger and G. Stein, ArXiv e-prints (2018), arXiv:1805.04537.
[24] H. M. Kamdar, M. J. Turk, and R. J. Brunner, MNRAS 455, 642 (2016), arXiv:1510.06402.
[25] S. Agarwal, R. Davé, and B. A. Bassett, MNRAS 478, 3410 (2018), arXiv:1712.03255.
[26] E. O. Nadler, Y.-Y. Mao, R. H. Wechsler, S. Garrison-Kimmel, and A. Wetzel, Astrophys. J. 859, 129 (2018), arXiv:1712.04467.
[27] C. Giocoli, M. Baldi, and L. Moscardini, ArXiv e-prints (2018), arXiv:1806.04681.
[28] W. Hu and I. Sawicki, Phys. Rev. D 76, 064004 (2007), arXiv:0705.1158.
[29] S. Hagstotz, M. Costanzi, M. Baldi, and J. Weller, ArXiv e-prints (2018), arXiv:1806.07400.
[30] J. Merten, C. Giocoli, M. Baldi, M. Meneghetti, A. Peel, F. Lalande, J.-L. Starck, and V. Pettorino, MNRAS submitted (2018).
[31] C. Giocoli, R. B. Metcalf, M. Baldi, M. Meneghetti, L. Moscardini, and M. Petkova, MNRAS 452, 2757 (2015), arXiv:1502.03442.
[32] J.-L. Starck, S. Pires, and A. Réfrégier, A&A 451, 1139 (2006), astro-ph/0503373.
[33] S. Pires, J.-L. Starck, A. Amara, A. Réfrégier, and R. Teyssier, A&A 505, 969 (2009).
[34] A. Leonard, S. Pires, and J.-L. Starck, MNRAS 423, 3405 (2012), arXiv:1204.4293.
[35] S. Pires, A. Leonard, and J.-L. Starck, MNRAS 423, 983 (2012), arXiv:1203.2877.
[36] A. Leonard, F. Lanusse, and J.-L. Starck, MNRAS 440, 1281 (2014), arXiv:1308.1353.
[37] F. Lanusse, J.-L. Starck, A. Leonard, and S. Pires, A&A 591, A2 (2016), arXiv:1603.01599.
[38] C.-A. Lin, M. Kilbinger, and S. Pires, A&A 593, A88 (2016), arXiv:1603.06773.
[39] A. Peel, C.-A. Lin, F. Lanusse, A. Leonard, J.-L. Starck, and M. Kilbinger, A&A 599, A79 (2017), arXiv:1612.02264.
[40] V. Nair and G. E. Hinton, in Proceedings of the 27th International Conference on International Conference on Machine Learning, ICML’10 (Omnipress, USA, 2010) pp. 807–814.
[41] D. P. Kingma and J. Ba, ArXiv e-prints (2014), arXiv:1412.6980.
[42] Y. Benjamini and Y. Hochberg, Journal of the Royal Statistical Society. Series B (Methodological) 57, 289 (1995).