

Probabilistic Object Classification using CNN ML-MAP layers

G. Melotti¹, C. Premebida¹, J. J. Bird², D. R. Faria², and N. Gonçalves¹

¹ ISR-UC, University of Coimbra, Portugal.
{gledson.melotti,cpremebida,nunogon}@isr.uc.pt
<https://www.isr.uc.pt>

² ARVIS Lab, Aston University, UK.
{birdj1,d.faria}@aston.ac.uk
<http://arvis-lab.io>

Abstract. Deep networks are currently the state-of-the-art for sensory perception in autonomous driving and robotics. However, deep models often generate overconfident predictions precluding proper probabilistic interpretation which we argue is due to the nature of the SoftMax layer. To reduce the overconfidence without compromising the classification performance, we introduce a CNN probabilistic approach based on distributions calculated in the network’s Logit layer. The approach enables Bayesian inference by means of ML and MAP layers. Experiments with calibrated and the proposed prediction layers are carried out on object classification using data from the KITTI database. Results are reported for camera (*RGB*) and LiDAR (range-view) modalities, where the new approach shows promising performance compared to SoftMax.

Keywords: Probabilistic inference, Perception systems, CNN probabilistic layer, object classification.

1 Introduction

In state-of-the-art research, the majority of CNN-based classifiers (Convolutional neural networks) train to provide normalized prediction-scores of observations given the set of classes, that is, in the interval $[0, 1]$ [1]. Normalized outputs aim to guarantee “probabilistic” interpretation. However, how reliable are these predictions in terms of probabilistic interpretation? Also, given an example of a non-trained class, how confident is the model? These are the key questions to be addressed in this work.

Currently, possible answers to these open issues are related to calibration techniques and penalizing overconfident output distributions [2–4]. Regularization is often used to reduce overconfidence, and consequently overfitting, such as the confidence penalty [4] which is added directly to the cost function. Examples of transformation of network weights include *L1* and *L2* regularization [5], Dropout [6], Multi-sample Dropout [7] and Batch Normalization [8]. Alternatively, highly confident predictions can often be mitigated by calibration tech-

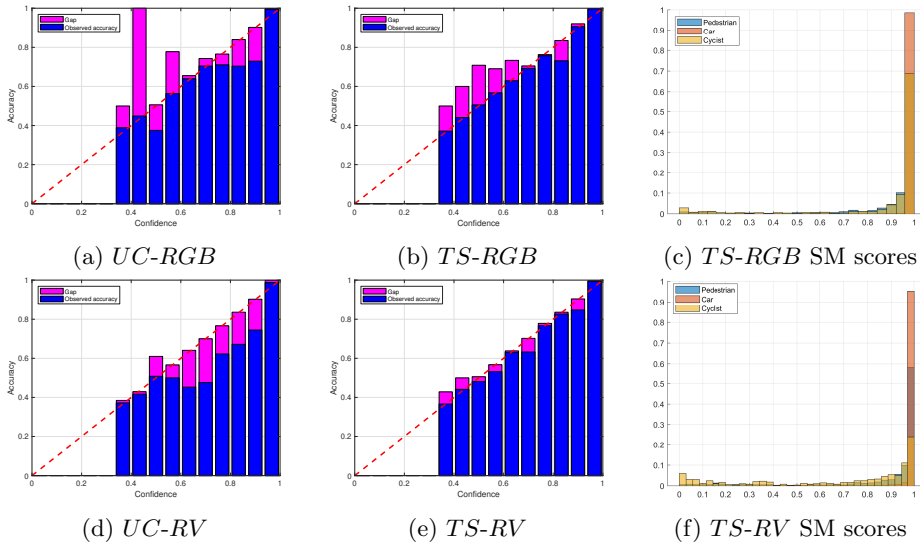


Fig. 1: RGB modality reliability diagrams (1st row), on the testing set, for uncalibrated (UC) in (a), and for temperature scaling (TS) in (b), with $T = 1.50$. Subfigure (c) shows the distribution of the calibrated prediction scores using SoftMax (SM). The 2nd shows the LiDAR (range-view: RV) modality reliability diagrams in (d) and (e), with $T = 1.71$, while in (f) is the prediction-score distribution. Note that (c) and (f) are still overconfident post-calibration.

niques such as Isotonic Regression [9] which combines binary probability estimates of multiple classes, thus jointly optimizing the bin boundary and bin predictions; Platt Scaling [10] which uses classifier predictions as features for a logistic regression model; Beta Calibration [11] which is the use of a parametric formulation that considers the Beta probability density function (pdf); and temperature scaling (TS) [12] which multiplies all values of the Logit vector by a scalar parameter $\frac{1}{T} > 0$, for all classes. The value of T is obtained by minimizing the negative log likelihood on the validation set.

Typically, post-calibration predictions are analysed via reliability diagram representations [3, 13], which illustrate the relationship the of the model’s prediction scores in regards to the true correctness likelihood [14]. Reliability diagrams show the expected accuracy of the examples as a function of confidence *i.e.*, the maximum SoftMax value. The diagram illustrates the identity function should it be perfectly calibrated, while any deviation from a perfect diagonal represents a calibration error [3, 13], as shown in Fig. 1a and 1b with the uncalibrated (UC) and temperature scaling (TS) predictions on the testing set. Otherwise, Fig. 1c shows the distribution of scores (histogram), which is, even after TS calibration, still overconfident. Consequently, calibration does not guarantee a good balance of the prediction scores and may jeopardize adequate probability interpretation.

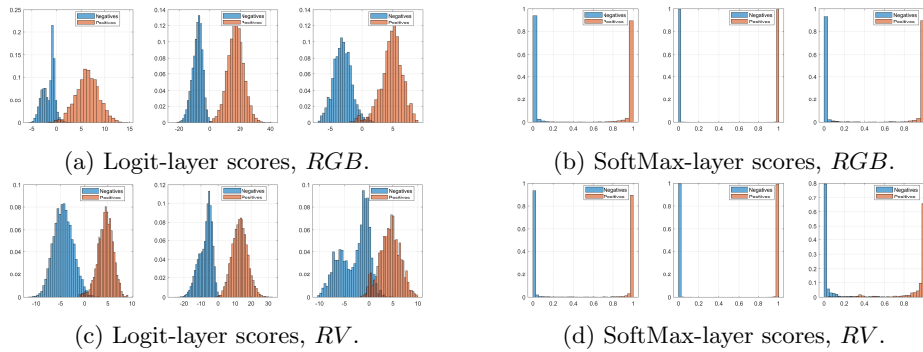


Fig. 2: Probability density functions (pdf), here modeled by histograms, calculated for the Logit-layer scores for *RGB* (a) and *RV* (c) modalities. The graphs in (a,b,c,d) are organized from left-right according to the examples on the Training set (where the positives are in orange). The distribution of the SoftMax prediction-scores in (b) and (d) are an evidence of high confidence.

Complex networks such as multilayer perceptron (MLPs) and CNNs are generally overconfident in the prediction phase, particularly when using the baseline SoftMax as the prediction function, generating ill-distributed outputs *i.e.*, values very close to zero or one [3]. Taking into account the importance of having models grounded on proper probability assumptions to enable adequate interpretation of the outputs, and then making reliable decisions, this paper aims to contribute to the advances of multi sensor (*RGB* and LiDAR) perception for autonomous vehicle systems [15–17] by using pdfs (calculated on the training data) to model the Logit-layer scores. Then, the SoftMax is replaced by a Maximum Likelihood (*ML*), or by a Maximum A Posteriori (*MAP*), as prediction layers, which provide a smoother distribution of predictive values. Note that it is not necessary to re-train the CNN *i.e.*, this proposed technique is practical.

2 Effects of Replacing the SoftMax Layer by a Probability Layer

The key contribution of this work is to replace the SoftMax-layer (which is a “hard” normalization function) by a probabilistic layer (a *ML* or a *MAP* layer) during the testing phase. The new layers make inference based on pdfs calculated on the Logit prediction scores using the training set. It is known that the SoftMax scores are overconfident (very close to zero or one), on the other hand the distribution of the scores at the Logit-layer is far-more appropriate to represent a pdf (as shown in Fig. 2). Therefore, replacement by *ML* or *MAP* layers would be more adequate to perform probabilistic inference in regards to permitting decision-making under uncertainty which is particularly relevant in autonomous driving and robotic perception systems.

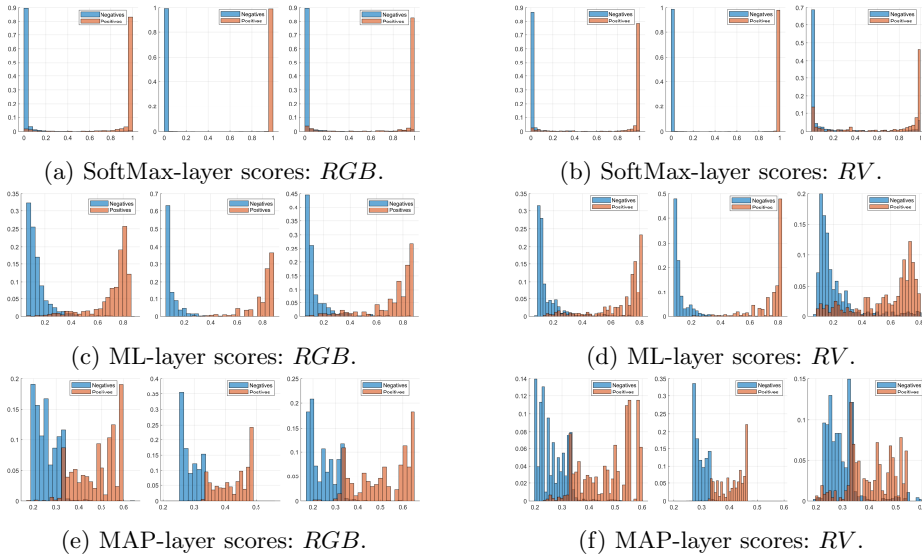


Fig. 3: Prediction scores (*i.e.*, the network outputs), on the Testing set, using SoftMax (baseline solution), ML and MAP layers, for *RGB* and LiDAR (*RV*) modalities.

Let X_i^L be the output score vector³ of the CNN in the Logit-layer for the example i , C_i is the target class, and $P(X_i^L|C_i)$ is the class-conditional probability to be modelled in order to make probabilistic predictions. In this paper, a non-parametric pdf estimation, using histograms with 25 (for the *RGB* case) and 35 bins (for the *RV* model), was applied over the predicted scores of the Logit-layer, on the training set, to estimate $P(X^L|C)$. Assuming the priors are uniform and identically distributed for the set of classes C , thus a *ML* is straightforwardly calculated normalizing $P(X_i^L|C_i)$, by the $P(X_i)$ during the prediction phase. Additionally, to avoid ‘zero’ probabilities and to incorporate some uncertainty level on the final prediction, we apply additive smoothing (with a factor equal to 0.01) before the calculation of the posteriors. Alternatively, a *MAP* layer can be used by considering, for instance, the *a-priori* as modelled by a Gaussian distribution, thus the i^{th} posterior becomes $P(C_i|X_i^L) = P(X_i^L|C_i)P(C_i)/P(X_i)$, where $P(C_i) \sim \mathcal{N}(\mu, \sigma^2)$ with mean μ and variance σ^2 calculated, per class, from the training set. To simplify, the rationale of using Normal-distributed priors is that, contrary to histograms or more tailored distribution, the Normal pdf fits the data more smoothly.

3 Evaluation and Discussion

³ The dimensionality of X is proportional to the number of classes.

Table 1: Classification performance (%) in terms of average F-score and FPR for the baseline (SM) models compared to the proposed approach of ML and MAP layers. The performance measures on the ‘unseen’ dataset are the average and the variance of the prediction scores.

Modalities:	SM_{RGB}	ML_{RGB}	MAP_{RGB}	SM_{RV}	ML_{RV}	MAP_{RV}
F-score	95.89	94.85	95.04	89.48	88.09	87.84
FPR	1.60	1.19	1.14	3.05	2.22	2.33
$Ave.Scores_{unseen}$	0.983	0.708	0.397	0.970	0.692	0.394
$Var.Scores_{unseen}$	0.005	0.025	0.004	0.010	0.017	0.003

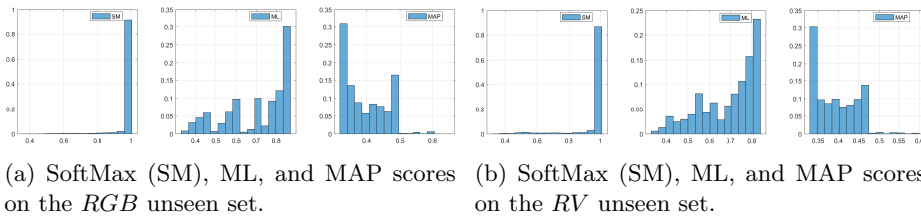


Fig. 4: Prediction scores, on the unseen data (comprising non-trained classes: ‘person_sit.’, ‘tram’, ‘trees/trunks’, ‘truck’, ‘vans’), for the networks using SoftMax-layer (left-most side), and the proposed ML (center) and MAP (right-side) layers.

In this work a CNN is modeled by Inception $V3$. The classes of interest are pedestrians, cars, and cyclists; the classification dataset is based on the KITTI $2D$ object [16], and the number of training examples are $\{2827, 18103, 1025\}$ for ‘ped’, ‘car’, and ‘cyc’. The testing set is comprised of $\{1346, 8620, 488\}$ examples respectively.

The output scores of the CNN indicate a degree of certainty of the given prediction. The “certainty level” can be defined as the confidence of the model and, in a classification problem, represents the maximum value within the SoftMax layer *i.e.*, equal to one for the target class. However, the output scores may not always represent a reliable indication of certainty with regards to the target class, especially when unseen or non-trained examples/objects occur in the prediction stage; this is particularly relevant for a real-world application involving autonomous robots and vehicles since unpredictable objects are highly likely to be encountered. With this in mind, in addition to the trained classes (‘ped’, ‘car’, ‘cyc’), a set of untrained objects are introduced: ‘person_sit.’, ‘tram’, ‘truck’, ‘vans’, ‘trees/trunks’ comprised of $\{222, 511, 1094, 2914, 45\}$ examples respectively. All classes with the exception of ‘trees/trunks’ are from the aforementioned KITTI dataset directly, while the former is additionally introduced by this study. The rationale behind this is to evaluate the prediction confidence of the networks on objects that do not belong to any of the trained classes,

and thus consequently the consistency of the models can be assessed. Ideally, if the classifiers are perfectly consistent in terms of probability interpretation, the prediction scores would be identical (equal to $1/3$) for all the examples on the unseen set on a per-class basis.

Results on the testing set are shown in Table 1 in terms of F-score metric and the average of the FPR prediction scores (classification errors). The average ($Ave.Scores_{unseen}$) and the sample-variance ($Var.Scores_{unseen}$) of the predicted scores are also shown for the unseen testing set.

To summarise, the proposed probabilistic approach shows promising results since ML and MAP reduce classifier overconfidence, as can be observed in Figures 3c, 3d, 3e and 3f. In reference to Table 1, it can be observed that the FPR values are considerably lower than the result presented by a SoftMax (baseline) function. Finally, to assess classifier robustness or the uncertainty of the model when predicting examples of classes untrained by the network, we consider a testing comprised of ‘new’ objects. Overall, the results are exciting since the distribution of the predictions are not extremities as can be observed in Fig. 4. Quantitatively, the average scores of the network using ML and MAP layers are significantly lower than the SoftMax approach, and thus are less confident on new/unseen negative objects .

References

1. Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. Is robustness the cost of accuracy? A comprehensive study on the robustness of 18 deep image classification models. In *ECCV*, 2018.
2. Ananya Kumar, Percy S Liang, and Tengyu Ma. Verified uncertainty calibration. In *NIPS*, pages 3792–3803. 2019.
3. Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 1321–1330, 2017.
4. Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. Regularizing neural networks by penalizing confident output distributions. CoRR, arXiv: 1701.06548, 2017.
5. Andrew Y. Ng. Feature selection, L1 vs. L2 regularization, and rotational invariance. ICML 04. Association for Computing Machinery, 2004.
6. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.
7. Hiroshi Inoue. Multi-sample dropout for accelerated training and better generalization. volume abs/1905.09788, 2019.
8. Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei, editors, *32nd ICML*, volume 37, pages 448–456, 2015.
9. Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. *ACM SIGKDD International Conference on KDD*, 2002.
10. John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Adv. Large Margin Classifiers*, 10, 2000.

11. Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *20th AISTATS.*, pages 623–631, 2017.
12. Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS*, 2015.
13. Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. In *Advances in Neural Information Processing Systems 32*, pages 12316–12326. 2019.
14. Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *ICML*, pages 625–632, 2005.
15. Manuel Martin, Alina Roitberg, Monica Haurilet, Matthias Horne, Simon Reiss, Michael Voit, and Rainer Stiefelhagen. Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles. In *ICCV*, 2019.
16. A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research (IJRR)*, 32(11), 2013.
17. G. Melotti, C. Premebida, and N. Gonalves. Multimodal deep-learning for object recognition combining camera and LIDAR data. In *IEEE ICARSC*, 2020.