Physical Data Models in Machine Learning Imaging Pipelines

Marco Aversa
University of Glasgow and Dotphoton AG
Glasgow, United Kingdom
marco.aversa@glasgow.ac.uk

Luis Oala
Fraunhofer HHI
Berlin, Germany
luis.oala@hhi.fraunhofer.de

Christoph Clausen
Dotphoton AG
Zug, Switzerland
christoph.clausen@dotphoton.com

Roderick Murray-Smith
University of Glasgow
Glasgow, United Kingdom
roderick.murray-smith@glasgow.ac.uk

Bruno Sanguinetti
Dotphoton AG
Zug, Switzerland
bruno.sanguinetti@dotphoton.com

Abstract

Light propagates from the object through the optics up to the sensor to create an image. Once the raw data is collected, it is processed through a complex image signal processing (ISP) pipeline to produce an image compatible with human perception. However, this processing is rarely considered in machine learning modelling because available benchmark data sets are generally not in raw format. This study shows how to embed the forward acquisition process into the machine learning model. We consider the optical system and the ISP separately. Following the acquisition process, we start from a drone and airship image dataset to emulate realistic satellite raw images with on-demand parameters. The end-to-end process is built to resemble the optics and sensor of the satellite setup. These parameters are satellite mirror size, focal length, pixel size and pattern, exposure time and atmospheric haze. After raw data collection, the ISP plays a crucial role in neural network robustness. We jointly optimize a parameterized differentiable image processing pipeline with a neural network model. This can lead to speed up and stabilization of classifier training at a margin of up to 20% in validation accuracy.

1 Introduction

In most common supervised-learning models for computational imaging, the typical approach is to feed the model with already processed images, extrapolate information and obtain a result depending on the model's task. Consequently, the model strictly depends on the data distribution, learning its spatial features and noise model statistics. However, [17] points out that perturbations applied to an already processed image can produce artefacts that are not faithful to the physics of camera processing. Furthermore, results in optics further support the concern that the noise obtained from an image processing pipeline is distinct from noise added to an already processed image [18, 9]. Taking a step back, the data which better resemble the object's physical properties is the raw image collected directly on the sensor. Data from all current sensors is processed first in the analogue and then in the digital domain. This processing, however, if done with care, does not detract from the "rawness"

Machine Learning and the Physical Sciences workshop, NeurIPS 2022.

of the data. Every image signal processing (ISP) pipeline introduces slight variations to the image, irreversibly corrupting the original information.

In this study, we embedded physics prior knowledge into a machine learning model starting from the object, through the optics, to the sensor. The aim is to open neural networks' black box by combining it with a well-known forward process, the white-box model. Having access to a well-defined white-box model, we can embed its information inside the network in order to obtain a hybrid model where we partially know how it should respond. The framework proposed is a hybrid model which combines our physical knowledge of the optical setup, sensor calibration data and ISP with a less interpretable learning process. Combining enables us to go beyond what is possible with augmentation [7, 5, 14] and catalogue testing [10, 2, 13, 11].

Our main contributions are:

- **Parametric ISP** We embedded a parametric mathematical data model into the model architecture. Given this differentiable forward model ϕ_{proc}, the gradient from the upstream task model ϕ_{task} can propagate to ϕ_{proc}. Thus, it enables the model to adapt to a specific white-box forward process. The data model considered for this application is an image signal process pipeline.
- **Satellite Imagery Emulation** We provide an explicit end-to-end parametric physics-based emulation. Starting from the drone imagery, based on the optical setup, sensor calibration properties and satellite dynamics, we emulated physically faithful satellite images. The process is differentiable and can be integrated into the model as ϕ_{proc}.

2 Physics-based data processing

What is a raw image Image acquisition has traditionally been optimized for the human perception of a scene [8, 16]. Hence imaging cameras, are usually calibrated to aid the human eye to perform some downstream task. However, this process that gives rise to optical image data, which ultimately forms the basis for downstream machine learning models, is rarely considered in the machine learning robustness literature. Conversely, most research has been conducted on processed RGB image representations. The *raw sensor image* x_{RAW} obtained from a camera differs substantially from the processed image that is used in conventional machine learning pipelines. A more precise term for raw data would be "metrologically accurate" data, and it preserves the following properties:
1. The statistical uncertainty (also referred to as error or noise) of a given pixel is uncorrelated to other pixels.
2. Each sample arises from a well-defined statistical distribution.
3. Pixels are unbiased, i.e. mean sample values accurately represent the amount of incident light.
These are necessary to apply standard statistical methods to data. The x_{RAW} state appears like a grey scale image with a grid structure. This grid is given by a colour filter array, commonly the Bayer pattern, which lies over sensors [4]. The final *RGB image* $\boldsymbol{v}$ is the result of a series of transformations applied to x_{RAW}. For many steps in this process, different possible algorithms exist. Starting from a single $\boldsymbol{x}_{RAW}$, all those possible combinations can generate an exponential number of possible images that are slightly different in terms of colours, lighting and blur – variations that contribute to dataset drift. In 1 a conventional pipeline from x_{RAW} to the final RGB image $\boldsymbol{v}$ is depicted.

Raw dataset acquisition As scientifically calibrated and labelled raw data is, to the best of our knowledge, currently not publicly available, we acquired two raw datasets as part of this study: Raw-Microscopy and Raw-Drone [12] [1]. Raw-Microscopy consists of expert annotated blood smear microscope images. Raw-Drone comprises drone images with annotations of cars. Our motivation behind the acquisition of these particular datasets was threefold. First, we wanted to ensure that the acquired datasets provide good coverage of representative machine learning tasks, including classification (Raw-Microscopy) and regression (Raw-Drone). Second, we wanted to collect data on applications that, to our minds, are disposed toward positive welfare impact in today's world, including medicine (Raw-Microscopy) and environmental surveying (Raw-Drone). Third, the optical and sensor forward model for a classical microscopy setup and drone has some similarities. We took advantage of this architecture to build a general emulation framework.

[1] We make both datasets publicly available at https://zenodo.org/record/5235536

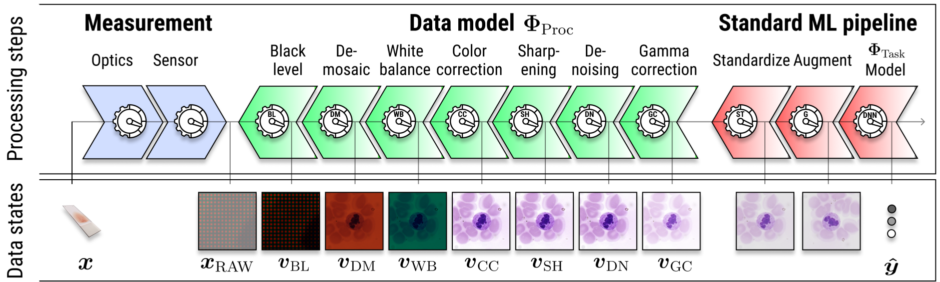

Figure 1: Schematic illustration of an optical imaging pipeline, the data states and novel, raw-enabled dataset drift controls. Data x transitions through different representations. The measurement process yields metrologically accurate raw data x_{RAW}, where the errors on each pixel are uncorrelated and unbiased. From the raw sensor, state data undergoes stages of image signal processing (ISP) ϕ_{proc}, the data model we consider here. Finally, the data is consumed by a machine learning task model ϕ_{task} which outputs y.

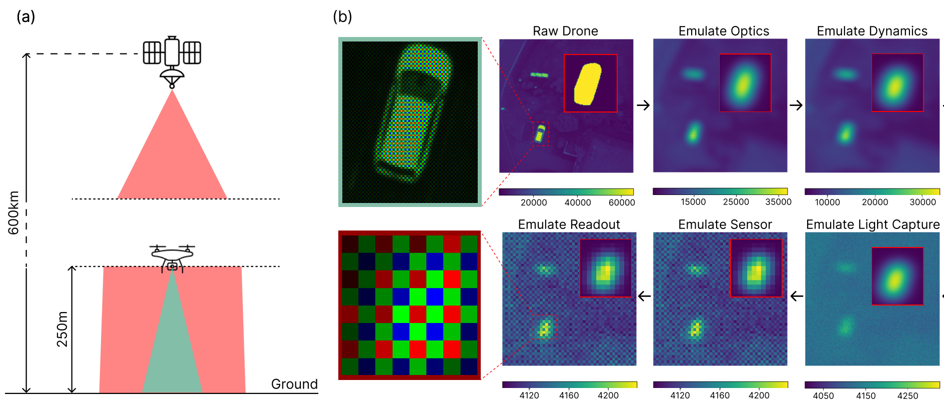

Figure 2: Metrologically consistent imagery satellite emulation. a) Schematic representation of the experimental setup. Raw images are collected with a drone flying at 250m. The emulated system is a payload satellite at the height of 600km. b) Sample false-colour images and segmentation masks (insets) at various stages of the emulation pipeline. The input image from the drone (top) is sharp, highly resolved and has good contrast, and the car (Bayer-pattern zoom on the top left) is easily recognized. The image is more and more degraded throughout the pipeline (images to the right and below), and only a featureless blob remains of the car (Bayer-pattern zoom on the bottom left).

3 Methods

Emulated Satellite The proposed method emulates the acquisition process of an imaging satellite payload starting from images collected with a drone [3]. A sketch of the emulated system is shown in Fig. 2a. As a use case, the end-to-end emulation pipeline is built to mimic STREEGO satellite's sensor and optical acquisition properties [15, 1]. Fig. 2b shows a schematic representation of the end-to-end process. The following will refer to the satellite system as *target* and the drone system as *source*. The emulation pipeline takes the optical model, the system dynamics details and the sensor properties of both the source and the target system. Given these ingredients, the emulation process follows the forward target model along all the data acquisition pipelines. We can control the output image quality through a set of parameters. Fixing a satellite's height and pixel size, the most significant parameters to control are the focal length of the optical system, the mirror's diameter and exposure time. The mirror diameter D affects the satellite's point spread function and collection efficiency. The diameter of the point spread function is inversely proportional to D, and the collection efficiency scales as D^2. A compromise must be made when choosing the exposure time. A longer exposure time improves the signal-to-noise ratio but introduces motion blur due to satellite motion. As a result, a point source is stretched out along the satellite trajectory. In a common satellite, the balance between the object magnification and the field of view depends on the focal

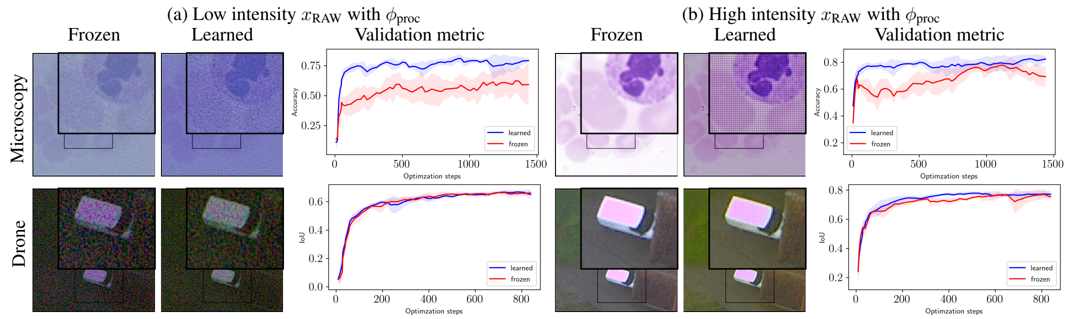

Figure 3: Low (a) and high (b) intensity images processed by a *frozen* and a *learned* pipeline. This dataset drift adjustment would not be possible with processed data typically used for machine learning experiments. The plots in the rightmost column of each block display the mean of validation metrics over five cross-validation runs. Error bars are reported as one standard deviation. Optimization steps 1439 and 915 correspond to epoch 60 into training.

length. Shortening focal lengths lead to a lower magnification and a broader angle of view. Since we can cover only a limited area with the drone, we estimated the corresponding downsampling to obtain the sub-image of the broader area covered by the satellite.

The parametrized data model The parametrized data model ϕ_{proc} maps a given raw image x_{RAW} to a RGB image. The data model is differentiable wrt. its parameters θ. This allows for backpropagation of the gradient from the output of the task model ϕ_{task} through the data model ϕ_{proc} back to the raw sensor image x_{RAW}. The forward model is defined as the composition of the most common ISP transformations $\phi_{proc} = \prod \phi_i(\theta_i)$ shown in Fig. 3. In the *learned* setting, the data model parameters are jointly optimized with the task model parameters. In the *frozen* setting, only the task model parameters are optimized, and the data model parameters are kept fixed.

4 Experiments

A parametrized data model ϕ_{proc} is paired with a task model ϕ_{task}. Experiments are performed on high and low-intensity images x_{RAW}, in-silico generated with a decreased exposure time and resampled with a calibrated noise model. In the left column (a) of Fig. 3 results on low intensity images are compared. The *learned* data model is better able to accommodate the dataset drift as visible in the improved stability of the learning trajectory. It exceeds that of the *frozen* data model (red line) by up to 25 percent in accuracy at a lower variance. In fact, the processed image from a *learned* data model (see *learned* column in block (a) of Fig. 3 for an example) can contain visible artifacts that *aid* stability and generalization vis-a-vis the image from the *frozen* baseline data model which, arguably, looks cleaner to the human eye. A possible explanation for the improved learning trajectory could be that a varying processing pipeline automatically generates samples akin to data augmentation. Such uses could be explored in scarce data settings like fine tuning, semi-supervised or few-shot learning. Having gradient access to the data model thus allows optimizing data generation for a given machine learning task. Suppose learned data models are to be applied in real-world applications. Thus, it appears likely that a tradeoff has to be made between human perceived visual quality and artefacts that can be helpful to the task model. For the segmentation task (bottom row of Fig. 3) the stabilization effect is not observable. This could be due to the low resolution of the problem itself, as the processing may not have a large effect on enhancing the solid blocks of cars in the raw data, as well as evidence suggesting that inverse problems are inherently less unstable [6]. Similar outcomes for stability and artefacts can also be observed for the reverse situation (high intensity) in the right column (b) of Fig. 3).

5 Discussion and Future works

We present how parameterized data models can be used to control dataset drift under physical constraints. The differentiable forward model is optimized on a constrained range of parameters. Therefore, to generalize the framework to a different, forward model, it is necessary to impose a prior on the parameters' physical boundaries for each transformation. Our experiment evaluated the model on high and low intensity emulated images by emulating the sensor exposure time. Going beyond physically faithful dataset drift controls, an interesting future extension to these experiments includes training directly jointly with the emulation process to find the optimal optical and sensor parameters for the downstream task model. Therefore, the metrologically accurate emulation can be used as prior before the payload is physically built. For example, one could consider using the emulated images for to optimize a super-resolution algorithm and then deploy the model on real satellite data.

6 Broader Impact

We propose an approach to embed optics, sensor and image processing prior knowledge into a neural network model. The differentiable forward acquisition process is jointly optimized with the task model. The forward model is controlled with physically constrained parameters. The prior knowledge of the physics system is embedded through this parameters distribution. Adapting the forward model, let the model explore this parameter space to find the optimal parameter configuration for the downstream task model. Furthermore, the framework allows freezing the task model while optimizing the forward model and vice versa. It is possible to use the framework to evaluate a pre-trained model on several slightly different processing pipelines. A usage scenario is the prospective validation of their task model to drift from different camera devices, for example, microscopes across different labs, without having to collect measurements from the different devices. Furthermore, the optics and sensor emulation can be deployed to make the task model resilient to different qualities of imaging systems. For example, hospital imaging setups have not improved for decades. Suppose a model is trained on the existing setup. If other images from better systems are added in the future, they can be mapped back to the same quality this was trained on. This method allows a system developed for a high-quality sensor to be verified also on older, lower quality systems.

References

[1] Mojtaba Abolghasemi and Dariush Abbasi-Moghadam. "Design and performance evaluation of the imaging payload for a remote sensing satellite". In: *Optics Laser Technology* 44.8 (2012), pp. 2418–2426. ISSN: 0030-3992. DOI: https://doi.org/10.1016/j.optlastec.2012.04.006. URL: https://www.sciencedirect.com/science/article/pii/S0030399212001600.

[2] B Albertina et al. "Radiology data from the cancer genome atlas lung adenocarcinoma [tcga-luad] collection". In: *The Cancer Imaging Archive* (2016).

[3] Marco Aversa et al. "Data-centric AI workflow based on compressed raw images". In: (2022). DOI: 10.5281/zenodo.7244932.

[4] Bryce E Bayer. *Color imaging array*. US Patent 3,971,065. July 1976.

[5] Phillip Chlap et al. "A review of medical image data augmentation techniques for deep learning applications". In: *Journal of Medical Imaging and Radiation Oncology* 65 (June 2021). DOI: 10.1111/1754-9485.13261.

[6] Martin Genzel, Jan Macdonald, and Maximilian Marz. "Solving Inverse Problems With Deep Neural Networks - Robustness Included". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022), pp. 1–1. DOI: 10.1109/TPAMI.2022.3148324.

[7] Dan Hendrycks and Thomas Dietterich. "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations". In: *Proceedings of the International Conference on Learning Representations* (2019).

[8] Robert William Gainer Hunt and Michael R Pointer. *Measuring colour*. John Wiley & Sons, 2011.

[9] Ronnachai Jaroensri et al. "Generating Training Data for Denoising Real RGB Images via Camera Pipeline Simulation". In: *arXiv* 1904.08825 (2019).

[10] Pang Wei Koh et al. "WILDS: A Benchmark of in-the-Wild Distribution Shifts". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 18–24 Jul 2021, pp. 5637–5664. URL: https://proceedings.mlr.press/v139/koh21a.html.

[11] Weixin Liang and James Zou. "MetaShift: A Dataset of Datasets for Evaluating Contextual Distribution Shifts and Training Conflicts". In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=MTex8qKavoS.

[12] Luis Oala and Marco Aversa et al. "Data Models for Dataset Drift Controls in Machine Learning With Images". In: (2022). DOI: 10.48550/ARXIV.2211.02578. URL: https://arxiv.org/abs/2211.02578.

[13] C Matek et al. "A single-cell morphological dataset of leukocytes from AML patients and non-malignant controls (AML-Cytomorphology_LMU)". In: *The Cancer Imaging Archive (TCIA)* (2019).

[14] Christian Matek and Carsten Marr. "Robustness evaluation of a Convolutional Neural Network for the classification of single cells in Acute Myeloid Leukemia". In: *ICLR 2021, RobustML workshop*. 2020.

[15] Massimiliano Rossi et al. "Fabrication and testing of STREEGO: a compact optical payload for earth observation on small satellites". In: *Optical Systems Design 2015: Optical Fabrication, Testing, and Metrology V*. Vol. 9628. SPIE. 2015, pp. 8–16.

[16] Andy Rowlands. *Physics of digital photography*. IOP Publishing, 2017.

[17] Rohan Taori et al. "Measuring Robustness to Natural Distribution Shifts in Image Classification". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2020. URL: https://arxiv.org/abs/2007.00644.

[18] GJJ Verhoeven. "It's all about the format–unleashing the power of RAW aerial photography". In: *International Journal of Remote Sensing* 31.8 (2010), pp. 2009–2042.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section 2
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Data and code are provided through an URL
(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [No]
(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Fig. 3
(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]