Dirty Pixels: Towards End-to-End Image Processing and Perception

STEVEN DIAMOND*, Stanford University
VINCENT SITZMANN*, Stanford University, MIT
FRANK JULCA-AGUILAR*, Algolux
STEPHEN BOYD, Stanford University
GORDON WETZSTEIN, Stanford University
FELIX HEIDE, Princeton University

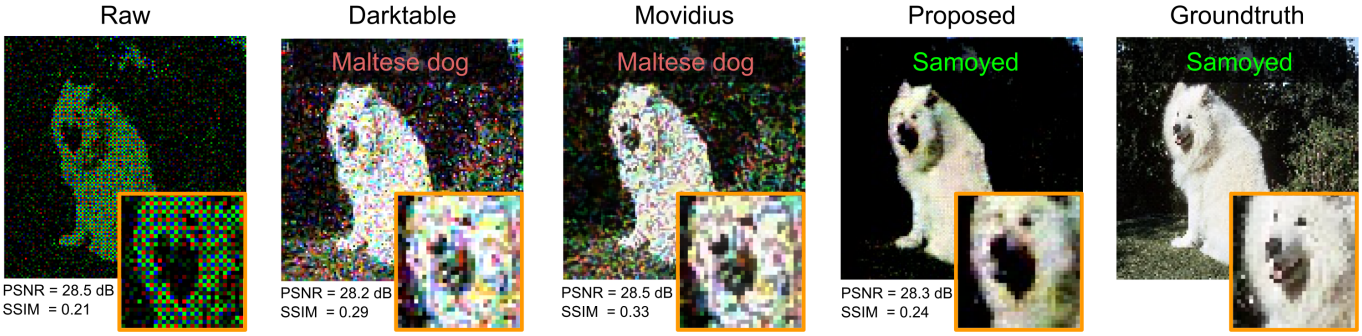

Fig. 1. A RAW input image subsampled on a color filter array and corrupted by sensor characteristics in low light (left) and its class prediction using MobileNet-v1 along with conventional processing pipelines. Processing RAW data using conventional image processing pipelines (ISPs) does not necessarily improve performance because conventional pipelines are optimized for human viewing, not for machine vision. Here, the image of a Samoyed dog is missclassified as the much smaller Maltese dog with thinner coat and smaller snout. We propose an end-to-end architecture for joint demosaicking, denoising, deblurring, and classification that makes classification robust in low-light scenarios. The proposed architecture learns a processing pipeline optimized for classification, which enhances fine details relevant for this high-level task – at the expense of more noise as measured by conventional metrics, PSNR and SSIM – and improves state-of-the art accuracy. Here, the dog's snout, ears, fur and outline are enhanced in contrast at the loss of surrounding background class regions. The proposed architecture has a principled and modular design and generalizes across light levels and cameras.

Real-world imaging systems acquire measurements that are degraded by noise, optical aberrations, and other imperfections that make image processing for human viewing and higher-level perception tasks challenging. Conventional cameras address this problem by compartmentalizing imaging from high-level task processing. As such, conventional imaging involves processing the RAW sensor measurements in a sequential pipeline of steps, such as demosaicking, denoising, deblurring, tone-mapping and compression. This pipeline is optimized to obtain a visually pleasing image. High-level processing, on the other hand, involves steps such as feature extraction, classification, tracking, and fusion. While this silo-ed design approach allows for efficient development, it also dictates compartmentalized performance metrics, without knowledge of the higher-level task of the camera system. For example, today's demosaicking and denoising algorithms are designed using perceptual image quality metrics but not with domain-specific tasks such as object detection in mind. We propose an end-to-end differentiable architecture that jointly performs demosaicking, denoising, deblurring, tone-mapping, and classification. The architecture does not require any intermediate losses based on perceived image quality and learns processing pipelines whose outputs differ from those of existing ISPs optimized for perceptual quality, preserving fine detail at the cost of increased noise and artifacts. We show that state-of-the-art ISPs discard information that is essential in corner cases, such as extremely low-light conditions, where conventional imaging and perception stacks fail. We demonstrate on captured and simulated data that our model substantially improves perception in low light and other challenging conditions, which is imperative for real-world applications like autonomous driving, robotics, and surveillance. Finally, we found that the proposed model also achieves state-of-the-art accuracy when optimized for image reconstruction in low-light conditions, validating the architecture itself as a potentially useful drop-in network for reconstruction and analysis tasks beyond the applications demonstrated in this work. Our proposed models, datasets, and calibration data are available at https://github.com/princeton-computational-imaging/DirtyPixels

CCS Concepts: • **Computing methodologies** → **Computer vision**; **Image processing**; **Supervised learning by classification**; **Neural networks**.

Additional Key Words and Phrases: computational photography, machine learning

*These authors contributed equally to this research.

Authors' addresses: Steven Diamond, Stanford University, diamond@cs.stanford.edu; Vincent Sitzmann, Stanford University, MIT, sitzmann@mit.edu; Frank Julca-Aguilar, Algolux, frank.julca-aguilar@algolux.com; Stephen Boyd, Stanford University, boyd@stanford.edu; Gordon Wetzstein, Stanford University, gordonwz@stanford.edu; Felix Heide, Princeton University, fheide@cs.princeton.edu.

0730-0301/2021/1-ART1 $15.00
https://doi.org/10.1145/3446918

ACM Reference Format:
Steven Diamond, Vincent Sitzmann, Frank Julca-Aguilar, Stephen Boyd, Gordon Wetzstein, and Felix Heide. 2021. Dirty Pixels: Towards End-to-End Image Processing and Perception. *ACM Trans. Graph.* 1, 1, Article 1 (January 2021), 15 pages. https://doi.org/10.1145/3446918

1 INTRODUCTION

Image sensor measurements are affected by various degradations in the physical image formation process. Raw sensor readings suffer from photon shot noise, optical aberration, read-out noise, spatial subsampling in the color filter array (CFA), spectral cross-talk on the CFA, motion blur, and other imperfections. The image signal processor (ISP) is a hardware block that addresses these degradations by processing the RAW measurement in a sequential pipeline of steps [Ramanath et al. 2005b] each targeting a sub-problem in isolation, before displaying or saving the resulting output image. The ISP performs an extensive set of operations, such as demosaicking [Zhang et al. 2011], denoising, deblurring, and tone-mapping. All of these low-level imaging tasks are ill-posed problems with recent active research [Chen et al. 2018; Gharbi et al. 2016; Heide et al. 2014; Zhang et al. 2016]. Existing image reconstruction algorithms are designed to minimize an explicit or implicit reconstruction loss aligned with human perceptions of image quality, as a prior to resolve the ill-posedness of the sub-problems listed above. Explicit losses are based on chart-based metrics [Phillips and Eliasson 2018], and emerging domain-specific standards, such as CPIQ [Jin et al. 2017], DxOMark, VCX Score for cellphone imaging and the emerging IEEE P2020 standard [Stead 2016] for autonomous vehicles. However, the approach widely adopted by ISP manufacturers is to design and tune ISPs to eliminate artifacts *human experts find visually unpleasant*, thereby minimizing an implicit perceptual loss.

At the same time, applications in emerging domains, including autonomous driving, robotics, and surveillance, consume images directly by a higher-level analysis module without ever being viewed by humans. Human expert assessment is not applicable to these "image-free" cameras, and this gives rise to the question if low-level processing is necessary, or if existing higher-level networks should better be trained directly on RAW sensor data.

ISPs are useful in that they map data from diverse camera systems into a common interface, a visually pleasing image, that most large-scale computer vision datasets adopt, *e.g.*, [Deng et al. 2009; Lin et al. 2014]. For downstream tasks, the real-world performance of a deployed high-level network will be close to the performance on clean images so long as the low-level pipeline can approximately recover the latent clean image from RAW data. However, in challenging capture conditions, *i.e.*, the corner cases of the ISP, recovering the latent image is extremely challenging, such as low-light captures that are heavily degraded by photon shot noise. For example, a denoising block that is optimized for perceptual quality will remove apparent chromatic noise, *e.g.*, the Movidius Myriad 2 ISP includes a Chroma-NLM stage for perceptual quality [Moloney et al. 2014], thereby destroying high-frequency color detail that could be exploited in the higher-level image analysis. Identical design trade-offs are found other key processing blocks, such as demosaicking, tone-mapping, and sharpening [Moloney et al. 2014].

An immediate solution for such failure modes appears to be removing the ISP completely and training the perception model directly on RAW measurement data. That way no information will be suppressed in the low-level image processing modules. Indeed, we demonstrate that existing classifiers trained on RAW data perform on-par with pre-processing from traditional ISPs, hand-crafted for perceptual viewing instead of CNN feature extraction.

In this work, we depart from traditional ISPs, and investigate learned architectures that perform end-to-end image processing and classification jointly. We propose an end-to-end differentiable model that uses RAW color filter array data as input and outperforms existing deep classification directly trained on this RAW input streams by a more than 5% in top-5 accuracy on in-the-wild captures. We validate that low light is indeed a failure mode for conventional computer vision systems that combine existing ISPs with existing high-level networks. We propose a novel neural architecture for joint denoising and demosaicking, dubbed "Anscombe networks", that we learn jointly with a high-level network and that exploits knowledge of the camera image formation model. We show that fine-tuning an Anscombe network with a high-level model performs better than training a high-level model directly on the RAW data or on the output of traditional ISPs, or recent state-of-the-art learnable ISP [Chen et al. 2018]. We demonstrate that the proposed Anscombe network ISP generalizes across imaging setting akin to a traditional ISP. Nevertheless, the output of the neural ISP differs from that of traditional ISPs, scoring worse on traditional perceptual metrics when trained for classification. However, when trained for human viewing, and no downstream analytic task, the proposed architecture achieves state-of-the-art image quality for low-light imaging, highlighting the potential of domain-specific imaging pipelines.

The contributions of this paper are the following:

- We demonstrate that conventional perception pipelines, which use a state-of-the-art ISP and classifier trained on a standard JPEG dataset, perform poorly in low light.
- We introduce Anscombe networks, a light-weight neural camera ISP for demosaicking and denoising that generalizes across camera architecture and capture settings. We show that Anscombe networks, by themselves, achieve state-of-the-art image quality when trained for low-light imaging using a perceptual loss for image quality.
- We demonstrate that jointly learning Anscombe networks with classification networks outperform training the high-level networks directly on RAW data or the output of state-of-the-art software, hardware and learnable ISPs, both when trained from scratch or fine-tuned.
- We evaluate the joint end-to-end model on synthetic and captured RAW data. To this end, we introduce a dataset of realistic noise and blur models calibrated from mobile cameras and a dataset of annotated noisy RAW captures.
- We demonstrate a real-time smart-phone implementation of the proposed end-to-end low-light classification model.

In the future, a large portion of our images will be consumed by high-level perception stacks, not by humans. We propose to reexamine the foundational assumptions of image processing (ISPs). Existing approaches tackle this challenge either by discarding ISPs

and retraining downstream networks directly on RAW data, or they manually tune, or optimize the parameters of hardware ISPs for a fixed network. Our work departs from both approaches and, to the best of our knowledge, it is the *first that jointly learns image processing (ISP) and classification network parameters* in an end-to-end fashion. We note that *specialized domain-specific processing is the goal of the proposed approach*. We do not dismiss traditional ISPs for general imaging tasks with unknown downstream applications but illustrate the potential of domain-specific camera processing.

2 RELATED WORK

Effects of Noise and Blur on High-level Networks. A small body of work has explored the effects of noise and blur on deep networks trained for high-level vision tasks. Dodge and Karam evaluated a variety of recent classification networks under noise and blur and found a substantial drop in performance [2016]. Vasiljevic et al. similarly showed that blur decreased classification and segmentation performance for deep nets, though much of the lost performance was regained by fine-tuning on blurry images [2016]. Karahan et al. showed that noise and blur degrade the performance of CNNs trained for face recognition [2016]. Several authors demonstrated that preprocessing noisy images with trained or classical denoisers improves the downstream performance [Agostinelli et al. 2013; da Costa et al. 2016; Jalalvand et al. 2016; Tang and Eliasmith 2010; Tang et al. 2012]. Chen et al. showed that training a model for denoising and separately classification can improve performance on both tasks [2016] when tested on corrupted versions of the MNIST and USPS datasets. Note that the models *trained from scratch*, in Tables 1 and 2, are equivalent to Chen et al. [2016] approach, where we optimize the classification network directly from RAW data.

Camera Image Processing Pipelines. Most digital cameras perform low-level image processing such as denoising and demosaicking in a hardware ISP pipelines based on efficient heuristics [Ramanath et al. 2005a; Shao et al. 2014; Zhang et al. 2011]. Modern imaging systems for cellphone use-cases may acquire a burst of images or images from multiple camera modules. Recently, Hasinoff et al. [2016] have demonstrated high-quality imaging in low light using bursts, which are then processed in a software ISP tuned for perceptual quality. Cameras for driver assistant systems, autonomous cars or other robotic purposes, however, have to react in real-time and therefore cannot acquire sequential exposures, leading to the emerge of split-pixel sensors (OmniVision OV10640, OV10650) and domain specific ISPs, such as the ARM Mali C71. Most conventional camera ISPs are implemented as fixed-function ASIC blocks to handle high-resolution image feeds at real-time rates [MT9P111 2015]. Only recently, camera ISPs are starting to become more programmable (also the case for software ISPs such as Hasinoff et al. [2016]). The Movidius Myriad 2 [Moloney et al. 2014] hardware ISP offers configurable pipelines with room for a few general-purpose blocks run on SIMD Vector Processors, but still relies on a large number of fixed-function hardware blocks. Hegarty et al. [2014] propose a domain-specific language for camera ISP processing on FPGAs, which translates image processing pipelines into efficient, low-power FPGA architectures. Instead of designing pipelines, Heide et al. [2014] pose low-level image processing as an optimization problem, achieving higher quality than previous ISPs for a variety of camera systems. However, their iterative optimization method is computationally intensive and an order of magnitude slower than real-time. Recently, Gharbi et al. rely on deep convolutional architectures to perform low-level vision tasks, such as demosaicking [2016] or tonemapping [2017]. While being computationally efficient, their architectures depend on heavily engineered datasets for training their models, whereas we use standard classification datasets. Liba et al. [2019] proposed a system for capturing images in low-light conditions based on the alignment and combination of multiple frames, and learning-based white balance and tonemapping. Schwartz et al. [2019], Liang et al. [2019], and Chen et al. [2018] proposed learnable ISPs based on deep convolutional networks. The model proposed by Chen et al. [2018] consists of convolutional network with CFA pixel packing similar to [Gharbi et al. 2016]. While their results are perceptually on-par or better than naive post-filtering approaches, using BM3D [Dabov et al. 2007] as an artifact suppression block, it remains unclear if recent state-of-the-art ISPs using traditional denoising blocks on RAW data, i.e., not as post-processing artifact suppression block, perform better as concluded in [Plotz and Roth 2017]. Our results described in Section 5 show that our proposed Anscombe ISP improves accuracy of a classifier trained on top of Chen et al. [2018]-preprocessed (and finetuned) images.

Traditional Image Processing Pipelines for Computer Vision. The role of traditional hardware ISP components in vision systems was examined in [Buckler et al. 2017; Tseng et al. 2019; Yahiaoui et al. 2019]. Buckler et al. [2017] suggested that ISPs should be configurable to switch between a human-viewable mode and computer vision mode to produce data optimized for vision tasks. However, ISP parameter tuning by visual inspection is extremely challenging if performed manually, motivating simulation environments [Blasinski et al. 2018]. Simulated environments, unfortunately, suffer from a significant domain gap [Hoffman et al. 2017]. Recently Tseng et al. [2019] proposed an automatic method for optimizing black-box ISPs. They propose to model and learn a differentiable proxy function that approximates the entire image processing pipeline. In contrast to the proposed method, Tseng et al. rely on traditional hardware ISPs and optimize *only ISP hyperparameters, not the high-level network*. The efficacy of this approach relies on the accuracy of the ISP approximation. As such, in our low-light scenario, the approximator network from Tseng et al. failed, see Figure 5 Supplement. We note that none of the above methods propose a *jointly end-to-end optimized ISP and downstream network*.

Domain Adaptation. A common problem in deep neural networks trained for high-level computer vision tasks is domain shift, meaning the difference in image statistics between the training data and the unknown real-world data, leading to poor performance of a trained model in the final real-world scenario. The literature on domain adaptation includes many methods for adapting models trained on one distribution to a target distribution, ranging in sophistication from simply fine-tuning the model on labeled data from the target distribution to more recent work that only requires sparsely labeled or unlabeled data (*e.g.*, [Ganin and Lempitsky 2015; Long et al. 2015; Sun and Saenko 2016; Tzeng et al. 2017, 2014]). The

Fig. 2. A RAW frame captured indoors using a Nexus 6 rear camera (after demosaicking). The image was taken at ISO 3000 with a 32 ms exposure time. The noise in the image is clearly visible.

domain adaptation literature has an implicit assumption, however, that the mapping from the training domain to the target domain is unknown. In our problem of classification under noise and blur, the mapping from the clean training data to degraded real world data can be modeled extremely accurately. We are thus able to map the clean training data into the target domain through simulation, and moreover to efficiently incorporate our a-priori knowledge of the physical model into the classification architecture. We put this into practice in the design of our efficient Anscombe networks, which, as validated in Section 5.1, outperform existing image processing layers when integrated and end-to-end trained.

3 CAMERA IMAGE FORMATION MODEL

3.1 Image formation

We consider the image formation $\mathcal{I}$ for a RAW sensor image as

$$\begin{aligned} y_x &\sim \alpha\mathcal{P}\left(\sum_{c\in\{R,G,B\}} \mathbf{S}^c(k_c * \mathbf{E}_c x)/\alpha\right) + \mathcal{N}(0,\sigma^2) \\ \Leftrightarrow\ y_x &\sim \alpha\mathcal{P}\left(\mathbf{A}x/\alpha\right) + \mathcal{N}(0,\sigma^2) \\ y &= \mathcal{I}(x) = \Pi_{[0,1]}(y_x), \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^{3N}$ is the vectorized latent color image, with N being the number of pixels, $y \in \mathbb{R}^N$ is the measured RAW image, $\alpha > 0$ and $\sigma > 0$ are parameters in a Poisson and Gaussian distribution, respectively, the operator $\mathbf{E}_c$ extracts the color channel $c \in \{R, G, B\}$, k_c represents the lens point spread function (PSF) in the color channel c, $*$ denotes the linear operator corresponding to 2D convolution on the vectorized input, and $\Pi_{[0,1]}$ denotes projection onto the interval $[0, 1]$. The matrix $\mathbf{S}^c$ models the spatial sub-sampling for color filter c on the color filter array of the sensor. This matrix is a diagonal sub-sampling matrix defined as

$$\mathbf{S}^c_{ii} = \begin{cases} 1 & \text{if pixel } t \text{ has color filter } i, \\ 0 & \text{else,} \end{cases} \tag{2}$$

The image formation model from above is composed of a linear part $\mathbf{A}x$, modeling all optical effects in the capture process with the matrix $\mathbf{A}$, and a non-linear sampling process according to the noise characteristics of the sensor. The measured image follows the physically accurate Poisson-Gaussian noise model with clipping described by Foi et al. [2009; 2008]. In the noise model, decreasing the light level increases α, but the dynamic range is kept constant by increasing the ISO, represented by multiplying $\mathcal{P}(\mathbf{A}x/\alpha)$ by α.

The image formation model from Eq. (1) is general and applicable to a variety of different camera architectures, ranging from traditional Bayer CFA cameras to interlaced HDR sensors, each covered by changing the linear forward model $\mathbf{A}$ according to the given camera architecture. We refer the reader to [Heide et al. 2014] for a variety of camera architectures this model supports. Note that, in contrast to [Gharbi et al. 2016; Heide et al. 2014], we assume a more accurate noise model, including the Poissonian component which is critical for the model accuracy in the low-flux regime.

3.2 Calibration

We calibrated the parameters α, and σ of the image formation model from Sec. 3.1 by acquiring calibration captures of a charts containing patches of different shades of gray (*e.g.*, [ISO 12233:2014 2014]) at various gains with auto-white-balance disabled. We then follow Foi et al. [2009] to estimate the unknown noise parameters. The photograph on the left in Fig. 3 shows our noise calibration setup. The center plot in Fig. 3 shows plots of $s(x) = \text{std}(\tilde{y})$ versus $E[\tilde{y}]$ and $\hat{s}(\hat{x}) = \text{std}(y)$ versus $E[y]$ for different ISO levels on a Nexus 6P rear camera. The parameters α and σ at a given light level are computed from the $s(x)$ and $\hat{s}(\hat{x})$ plots. The noise under our calibrated image formation model can be high. Fig. 2 shows a typical capture of a Nexus 6 rear camera in low light. This image was acquired for ISO 3000 and a 32 ms exposure time. The only image processing performed on this image was bi-linear demosaicking. The severe levels of noise present in the image demonstrate that low and medium light conditions represent a major challenge for imaging and computer vision systems. Note that particularly inexpensive low-end sensors will exhibit drastically worse performance compared to higher end smartphone camera modules.

In addition, we calibrated the optical aberrations k from Eq. 1 using a Bernoulli noise chart with checkerboard features, following Mosleh et al. [2015] for spatially-varying PSF calibration. The right plots in Fig. 3 show the PSF k for entire field-of-view of a Nexus 5 rear phone camera optic. An in-depth description of our calibration procedure is provided in the Supplemental Material. Alternative approaches to learned data generation for image reconstruction methods have been proposed in [Brooks et al. 2019; Jaroensri et al. 2019].

4 END-TO-END FRAMEWORK

In this section, we describe the proposed architecture for joint denoising, demosaicking, (deblurring,) and classification. We evaluate the joint architecture in Sec. 5, as well as ablated models where only the low-level or high-level pipeline is trained or a conventional ISP pipeline is used. We assess the performance of the proposed model both on the simulated data and on *captured RAW images*, to show that our simulated results carry over to real data.

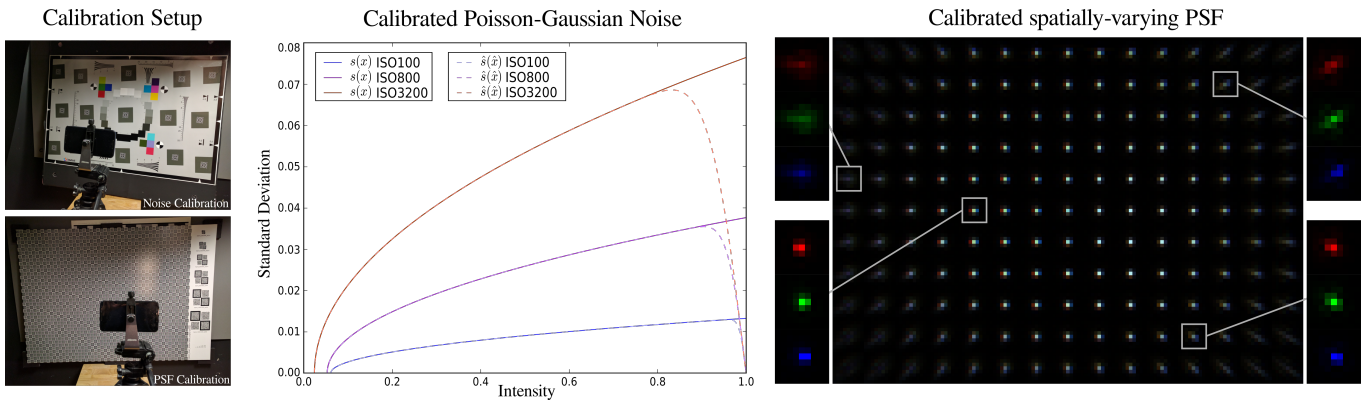

Fig. 3. (Top left) The noise calibration setup. (Bottom left) The PSF calibration setup. (Center) $s(x) = \text{std}(\tilde{y})$ versus $E[\tilde{y}]$ and $\hat{s}(\hat{x}) = \text{std}(y)$ versus $E[y]$ for different ISO levels on a Nexus 6P rear camera. The noise parameters α and σ at a given light level are computed from the $s(x)$ and $\hat{s}(\hat{x})$ plots. (Right) The PSFs for the entire field-of-view of a Nexus 5 rear camera. Two center PSFs, an off-axis PSF, and a periphery PSF are magnified.

The architecture proposed in this work is illustrated in Fig. 4. It combines jointly learned low-level and high-level processing units, taking RAW sensor CFA data as input and outputting image labels. We propose a *single differentiable model that generalizes across cameras and light levels*. This allows our model to abstract away the details of the camera for downstream applications, while being flexible and applicable to novel camera architectures.

We base the low-level block, which we dub Anscombe network unit, on an optimization algorithm Λ that solves the problem of reconstructing an uncorrupted latent mid-level representation from noisy, single-channel, spatially-subsampled RAW measurements. In contrast to standard CNN models, the Anscombe layers in this model make the approach light-level independent and the unrolled optimization model achieves generalization across camera models (without retraining). We express the joint reconstruction and perception problem as a bilevel optimization problem

$$\begin{aligned} &\min_{\vartheta, v} \mathcal{L}\left(\Lambda(y, \vartheta), x, v\right) \\ &\text{s.t. } \Lambda(y, \vartheta) = \underset{x}{\text{argmin}}\, \mathcal{G}\left(x, y, \vartheta\right), \end{aligned} \tag{3}$$

where Λ minimizes here a lower-level objective $\mathcal{G}$. The output layer of this lower-level unit is an multi-channel mid-level representation $\Lambda(y, \vartheta)$, which is input into the higher-level model component and associated classification loss $\mathcal{L}$. Here the model parameters v of the higher-level model are absorbed in $\mathcal{L}$ as a third argument.

For the nested objective $\mathcal{G}$, we follow a Bayesian approach as architecture backbone as it estimates a latent three-channel image x exploiting both the probabilistic image formation model and allows for priors expressed in a principled fashion. The Bayesian model assumes that x is drawn from a prior distribution $\Omega(\vartheta)$, parameterized by ϑ. We solve the Bayesian inference problem by unrolling an iterative optimization algorithm, only parameterizing the image prior with unknown, learned parameters, and truncating the iterations yielding the operator Λ.

Any differentiable higher-level image analysis method can be used in the proposed stack. In the following, we use the MobileNet-v1 classification network [Howard et al. 2017] as a higher-level network (which is replaced by a perceptual image loss when specializing the model for imaging for human vision 5.1). The higher-level classification loss $\mathcal{L}$ is the standard cross-entropy classification loss. We chose the MobileNet model family, since it is computationally efficient, running on modern smart-phone platforms in real-time, and while achieving competitive classification performance [Howard et al. 2017]. As the model is small, it can also be trained from scratch without data-center-scale training resources. Note that the proposed architecture can be adapted to other high-level computer vision tasks such as segmentation, object detection, and tracking, by replacing the classification network with another network for the given task. This also includes no high-level model, which then allows for the method to act as a learned ISP optimized for human viewing with adequate loss $\mathcal{L}$, which we demonstrate in Sec. 5.3.

4.1 Anscombe Networks

The proposed low-level image processing unit, Anscombe networks, performs image reconstruction as a statistical estimation problem, which estimates a feature-preserving mid-level image from corrupted observations. We adopt a Bayesian approach and derive the proposed Anscombe network model as a maximum-a-posteriori (MAP) estimation method. As part of this model we introduce novel Anscombe network layers in this section, which allow for an efficient, compact, and transferable model that hence behaves like an ISP but is differentiable. Central to the design of our Anscombe networks are our variance-stabilizing Anscombe transform layers. Anscombe layers map Poisson-Gaussian distributed measurements y, to IID Gaussian noise [Foi and Makitalo 2013] with variance $\sigma = 1$. Recall that the input to our Anscombe network, y, is the result of the camera image formation model defined in Eq. (1) (see calibration described in Section 3.2). We show in Section 5.1 that Anscombe networks improve the accuracy of classifiers, trained with RAW data, that use conventional layers with capacity similar to our Anscombe networks.

In the Bayesian model, an unknown latent image x is drawn from a prior distribution $\Omega(\vartheta)$ with parameters ϑ. The linear transform $\mathbf{A}$ from Eq. (1), modeling all optical processes, transforms x to the

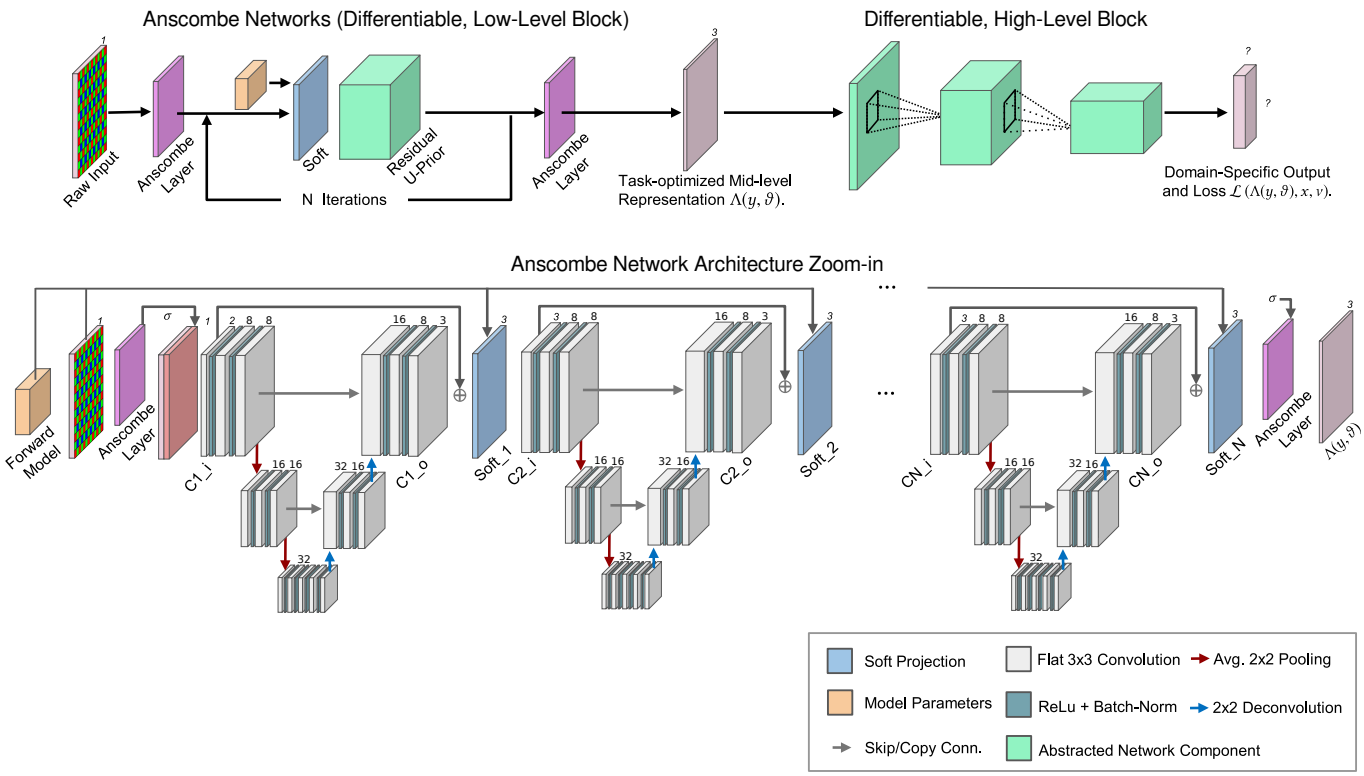

Fig. 4. The proposed end-to-end architecture (top) for joint denoising, demosaicking, (deblurring) and classification combines a novel low-level Anscombe network block and a high-level task-specific network component in a single stack that takes in RAW CFA sensor data and outputs image labels. The Anscombe network component (zoom-in on the bottom) exploits knowledge of the calibrated image formation model and a learned proximal operator in a recurrent manner. The high-level model takes the output of the Anscombe network unit (either a feature tensor or an image) and feeds it into a standard classification network trunk. This proximal operator in the Anscombe network is a recurrent residual U-Net model with dense skip connections across all operator "iterations". A partly unrolled network is show at the bottom.

incident signal on the sensor, which is then measured by this sensor as an image y drawn from a noise distribution $\omega(\mathbf{A}x)$. Recalling the image formation model from Eq. (1), the transform $\mathbf{A}$ models both the convolution with the kernel k and subsampling on the CFA, and ω represents the calibrated Poisson-Gaussian noise.

Then the posterior probability of an unknown image x yielding an observation y is

$$P(x|y;\vartheta) = \frac{P(y|\mathbf{A}x)P(x;\vartheta)}{\int_x P(y|\mathbf{A}x)P(x;\vartheta)} \tag{4}$$

with $P(y|\mathbf{A}x)$ being the probability of sampling y from $\omega(\mathbf{A}x)$ and $P(x;\vartheta)$ be the prior probability of sampling x from $\Omega(\vartheta)$. Because the posterior is proportional to $P(y|\mathbf{A}x)P(x;\vartheta)$ the MAP estimate of x is then given by

$$x = \operatorname*{argmax}_x P(y|\mathbf{A}x)P(x;\vartheta), \tag{5}$$

or equivalently

$$x = \operatorname*{argmin}_x \underbrace{f(y, \mathbf{A}x) + r(x, \vartheta)}_{\mathcal{G}(x,y,\vartheta)}, \tag{6}$$

where the data term $f(y, \mathbf{A}x) = -\log P(y|\mathbf{A}x)$ and prior $r(x, \vartheta) = -\log P(x;\vartheta)$ are negative log-likelihoods. These two terms define the lower-level objective $\mathcal{G}(x, y, \vartheta)$ from Eq. (3).

Implicit Unrolled Proximal Optimization. The low-level unit Λ minimizes the loss $\mathcal{G}$ by solving Eq. (6). A large variety of algorithms have been developed for solving problem (6) efficiently for different convex data terms and priors, *e.g.*, FISTA [Beck and Teboulle 2009b], Chambolle-Pock [Chambolle and Pock 2011], ADMM [Glowinski and Marroco 1975]). The majority of these algorithms are iterative methods, in which a mapping $\Gamma(x^k, \mathbf{A}, y, \vartheta) \rightarrow x^{k+1}$ is applied repeatedly to generate a series of iterates that converge to a solution $x^\star$, starting with an initial point x^0.

While an algorithm implementation can only be derived for explicitly given f and r, we can define the algorithm itself with implicitly defined objectives. Suppose f and r are convex in x, and r is differentiable. Then, we can solve Eq. (6) with the proximal gradient method [Beck and Teboulle 2009a,b; Diamond et al. 2017; Parikh

Algorithm 1 Anscombe networks: Variance-stabilized implicit proximal gradient network.

1: $\tilde{y}, \sigma \leftarrow \mathcal{A}(y)$
2: Recurrent vars: $x^0 = A^T y$, $\alpha_k = C_0 C^{-k}$, $C_0 > 0$, $C > 0$
3: **for** $k = 0$ to $N - 1$ **do**
4: $\quad x^{k+1/2} \leftarrow \text{CNN}(x^k, \vartheta^k)$.
5: $\quad x^{k+1} \leftarrow \text{argmin}_x \, \alpha_k f(Ax, \tilde{y}) + \frac{1}{2}\|x - x^k - x^{k+1/2}\|_2^2$.
6: **end for**
7: $\tilde{x} \leftarrow \mathcal{A}^{-1}(x^N, \sigma)$

and Boyd 2013], which consists of the following updates

$$x^{k+1/2} = x^k - \alpha_k \nabla_x r(x^k, \vartheta) \tag{7}$$

$$x^{k+1} = \underset{x}{\text{argmin}} \, f(y, \mathbf{A}x) + \frac{1}{2\alpha_k}\|x - x^{k+1/2}\|_2^2, \tag{8}$$

where $\alpha_k > 0$ is a step length. Each update consists of a prior step (7) and a data step (8). The data step (8) is known as the proximal operator of f, that is

$$\mathbf{prox}_{\frac{\lambda}{\beta} f(y, \mathbf{A}\cdot)}(x) = \underset{z}{\text{argmin}} \, \frac{\lambda}{\beta} f(y, \mathbf{A}z) + \frac{1}{2}\|x - z\|^2. \tag{9}$$

Please see [Parikh and Boyd 2013] for a detailed review of proximal operators and corresponding proximal optimization algorithms. One central idea that we rely on in this work is that we can also implicitly define steps of this algorithm. In particular, we propose to learn the prior mapping without explicitly defining the objective r, the space of all representations interpretable by the higher-level block, but rather parameterize the projection operator $\text{CNN}(v, \theta^k) = v - \alpha \nabla_x r(v, \vartheta)$ with ϑ and α being learned implicitly.

Solving Eq. (6) using an iterative optimization algorithm of the reader's choice would lead to an algorithm with a data-dependent termination criterion and no obvious method to learn unknown algorithm parameters since computing the derivatives of the output with respect to the algorithm parameters ϑ is value-dependent. An alternative approach is to execute a pre-determined number of iterations N, in other words unrolling the optimization algorithm. This approach is motivated by the fact that for many imaging applications very high accuracy, *e.g.*, convergence below tolerance of 10^{-6} for every local pixel state, is not needed in practice, as opposed to optimization problems in, for instance, control. Instead, many applications are runtime-constrained, and truncation allows for a fixed runtime. Fixing the number of iterations allows us to view the iterative method as an explicit function $\Gamma^N(\cdot, \mathbf{A}, y, \vartheta) \rightarrow x^N$ of the initial point x^0. Parameters such as ϑ may be fixed across all iterations or vary by iteration. The unrolled iterative algorithm can be interpreted as a deep network, and, if each iteration of the unrolled optimization is differentiable, the gradient of ϑ and other parameters with respect to a loss function on x^N can be computed efficiently through backpropagation. The proposed network recipe is given in Algorithm 1. Note that we allow all parameters to differ across layers. The model is differentiable in its output with respect to each layer's free parameters.

Anscombe Layers. The network generated by Algorithm 1 is an implicit unrolled proximal gradient network. However, rather than working directly on the measurements y, which are Poisson-Gaussian distributed according to Eq. (1), we embed the unrolled architecture in variance-stabilizing Anscombe transform layers, converting the Poisson-Gaussian noise into IID Gaussian noise [Foi and Makitalo 2013] with variance $\sigma = 1$. This has the benefit that the data step in line 5 of Alg. 1 becomes a simple quadratic term, and image features at all intensity levels are affected by the same noise degradations, effectively regularizing the model to perform robustly independent of the light level.

Specifically, we apply the generalized Anscombe transform [Foi and Makitalo 2013] as a first layer, denoted by the operator $\mathcal{A}$, to the measured single channel RAW observation y, The transform and its unbiased inexact inverse are defined as

$$\mathcal{A} : x \mapsto 2\sqrt{x + \tfrac{3}{8}}, \tag{10}$$

$$\mathcal{A}^{-1} : x \mapsto \frac{1}{4}x^2 - \frac{1}{8} - \sigma^2. \tag{11}$$

However, RAW data input to this transform, without modifications, results in peak signals that are not consistent across training examples. Hence, the gradient components of the unrolled proximal gradient method are not normalized with respect to light level, leading to poor network performance. To avoid this behavior, we max-normalize the output of the forward Anscombe transform with the multiplicative factor $s_{\mathcal{A}} = 1/\max(\mathcal{A}x)$. While this normalizes the value range to the interval $[0, 1]$, the unit-variance Gaussian noise distributed $\mathcal{A}x$ becomes Gaussian-distributed with variance $\sigma = s_{\mathcal{A}}^2$. As this parameter is known, we provide it to the network as a separate channel, which is illustrated in Fig. 4. The output of the unrolled proximal gradient network component followed by the inverse generalized Anscombe transform $\mathcal{A}^{-1}$, which inverts the shift and scaling, then applies the inverse transform. The noise parameters are known from the ISO and the precalibrated noise curves from Sec. 3.2.

Soft Projection Layers. The data step in Alg. 1 (line 5) is the "soft projection" operator $\Pi(\cdot, \gamma, \mathbf{A}, y)$ given by

$$\Pi(v, \gamma, \mathbf{A}, y) = \underset{z}{\text{argmin}} \, \frac{1}{2}\|y - \mathbf{A}z\|_2^2 + \frac{\gamma}{2}\|v - z\|_2^2.$$

Recalling Eq. (9), $\Pi(\cdot, \gamma, A, y)$ is the proximal operator of the function f. With the Anscombe layers present, this function, i.e. the negative log-likelihood $-\log P(y|\mathbf{A}x)$ from Eq. (6), becomes a simple quadratic now, that is

$$f(y, \mathbf{A}x) = \frac{1}{2}\|y - \mathbf{A}x\|_2^2.$$

Hence, the operator Π can be computed efficiently as an unconstrained quadratic optimization problem. In the case of joint demosaicking and denoising, the operator $\mathbf{A} = \mathbf{S}$ and Π becomes

$$\Pi(v, \gamma, \mathbf{S}, y) = \frac{\mathbf{S}^T \mathcal{A}y + \gamma z}{\mathbf{S}\mathbf{1} + \gamma},$$

where division is elementwise. The soft projection parameter $\gamma > 0$ trades off closeness to the input v with fidelity to the measurements y (*i.e.*, ensuring $y \approx Ax$). We dub this operator "soft projection" because in the limit $\gamma \rightarrow 0$, $\Pi(v, \gamma, A, y)$ is the Euclidean projection of v onto the linear system $y = \mathbf{A}x$. Note that γ may be learned or fixed.

The soft projection data step is inspired by analysis in the Supplemental Material. We found that substantially improved generalization over naive residual CNN models could be achieved due to applying soft projection in the proposed unrolled architecture, in particular for tasks where the imaging operator $\mathbf{A}$ varies across camera or example (for instance, different CFA patterns, optical systems, or images that are blurred with different blur kernels). Intuitively, soft projection decouples the (approximate) inversion of the physical image operator $\mathbf{A}$ from the prior step. Thus, the model does not have to re-learn the (approximate) inversion of $\mathbf{A}$ depending on sensor, optical parameters, or capture settings, and need instead only learn prior parameters and algorithm hyper-parameters.

4.2 Residual U-Net Prior Parametrization

The purpose of the cascade of prior network units in our architecture is to map estimates of the unknown midlevel representation x onto a nearby point in the manifold of representations that are interpretable by the higher-level network, or when optimizing for human viewing (i.e. the space of perceivable natural images). The prior steps from Algorithm 1 must therefore be flexible enough to learn the complex statistics of natural images but also project on a subset according to the higher-level loss $\mathcal{L}$.

We use a CNN as learnable prior architecture, as CNNs are established architectures for feature-encoding in the image domain and are thus a natural choice for learning the mapping onto the subset manifolds of natural images. Specifically, we propose a deep residual U-Net [Ronneberger et al. 2015] variant with three levels (see Figure 4), 3×3 convolution kernels, ReLU nonlinearities, downsampling with 2×2 average pooling, upsampling by 2×2 deconvolution layers (transpose convolution), and batch normalization in the intermediate layers to ease training [Ioffe and Szegedy 2015]. The number of channels in the first U-Net level increases by a factor of 2. The channels are doubled in each of the three levels of the U-Net. The U-Net prior at iteration k in the unrolled stack takes as input the output of the soft projection step $k-1$ concatenated with the Anscombe noise parameter σ as a separate channel. In order to handle the RAW color-filter array data, the very first layer in the U-Net prior at iteration 0 uses a stride 2 convolution in the very first convolutional layer. Further information on the U-Net parametrization can be found in the supplement.

We note that the U-Net priors are trained end-to-end as part of the complete architecture in Fig. 4 and a different prior is trained for each iteration of the unrolled optimization stack. This allows each prior step to specialize in removing *correlated noise*, i.e. reconstruction artifacts, introduced by the preceding data step, such as inpainting artifacts aligned with the CFA or inverse filtering ringing artifacts.

5 EVALUATION

Next, we describe the evaluation of our proposed methods. First, we evaluate our joint imaging and perception model on classification of low-light RAW images. Specifically, we captured a new dataset over a range of low-light levels and also built a synthetic low-light dataset based on ImageNet. We include ablated studies that show the importance of our proposed low-level Anscombe network to improve the high-level network accuracy. Second, we evaluate our low-level model for image reconstruction in low-light for human viewing (imaging without considering a high-level task, i.e. classification). For this evaluation, we use a recent publicly available dataset [Chen et al. 2018], which consists of short and low exposure images. Third, we demonstrate a real time mobile prototype implemented using the Android Camera2 API and a remote Tensorflow model server. The next sections describe the experimental setup and results found over these evaluations.

5.1 Evaluation of Low-light Imaging and Perception – Synthetic Data

We trained instances of the proposed joint architecture for four challenging scenarios: 3 lux, 6 lux, the range 2 to 20 lux, and the range 2 to 200 lux. While the first two settings allow us to analyze the models in specific low-light conditions, the scenarios with ranges of illuminance allow us to evaluate the generalization of the models over a variety of different light levels. Specifically, we trained and evaluated the models on a noisy version of ImageNet, constructed using the image formation model from Sec. 3.1, calibrated for the Nexus 5 rear camera for a given light-level (or a light-level sampled randomly from a range). To evaluate the effect of noise separately from optical aberrations, we ignore aberrations in the following (see Supplemental Material). The results reported next correspond to the ImageNet validation set of 50,000 images [Deng et al. 2009], which consists of 1000 object classes, and 50 samples per class.

To evaluate over many different noise settings and to be able to train deep nets completely from scratch (Table 1), we opt to use the computational efficient MobileNet classification network in all the following experiments. We refer the reader to the Supplementary document for results using the much larger Inception-v4 classifier on a smaller subset of the evaluations taking one month of training time. We compare the proposed joint architecture (Joint Anscombe Network and MobileNet-v1) to the following baselines:

- The conventional approach of combining a high-quality ISP, optimized for human viewing, with an existing pretrained MobileNet-v1 classifier.
- Using a trainable state-of-the-art ISP [Chen et al. 2018], finetuned for image quality in each noise scenario, and a MobileNet-v1 classifier finetuned on the learned ISP output.
- A MobileNet-v1 classifier directly trained from scratch on RAW noisy data.
- As a deeper version of our classifier with higher model capacity, we train the MobileNet-v2 (1.4) classifier [Sandler et al. 2018] from scratch, with 50% more parameters and about 40 million more FLOPS.

We train all the evaluated models until convergence with large iteration buffer. Table 1 summarizes the results for the described low-light scenarios. We next describe our findings from this evaluation.

Combining high-quality ISPs with pretrained high-level network fails in low-light. In this experiment, we use the hardware ISP of a Movidius Myriad 2 evaluation board, and the high-quality open-source ISP Darktable [2018] both engineered and optimized for visual image quality. We note that the Darktable uses a non-local means block-matching denoiser (NLM) [Buades et al. 2005] that

	3 lux		6 lux		2 to 20 lux		2 to 200 lux		Size and Complexity	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	# Params. (M)	FLOPs (M)
From Scratch MobileNet-v1 (RAW input w/o ISP)	23.53%	44.13%	32.65%	55.94%	35.16%	58.14%	42.65%	66.13%	4.23	181
From Scratch deeper MobileNet-v2 (RAW input w/o ISP)	27.87%	50.82%	36.80%	61.31%	36.32%	59.40%	38.56%	61.58%	6.90	320
Movidius Myriad 2 ISP + Pretrained	0.22%	1.10%	1.69%	5.39%	9.12%	18.63%	17.78%	32.11%	4.23	181
Darktable ISP + Pretrained	0.22%	0.46%	0.46%	1.52%	7.12%	15.28%	18.20%	32.04%	4.23	181
Movidius Myriad 2 ISP + Finetuning	23.52%	44.69%	36.11%	60.27%	17.31 %	34.98 %	21.75%	40.93 %	4.23	181
Darktable ISP + Finetuning	20.02%	39.56%	34.73%	58.55%	16.45 %	33.13%	23.47%	43.77%	4.23	181
U-Net [Chen et al. 2018] + Finetuning	29.89 %	52.62 %	36.20%	60.23%	14.38%	30.44%	20.23%	39.35%	11.99	537
U-Net [Chen et al. 2018] + percep. loss + Finetuning	9.52%	20.84%	25.74%	45.81%	19.09%	36.26%	26.17%	46.78%	11.99	537
Proposed Joint Architecture (MobileNet-v1 Head)	**30.50%**	**53.28%**	**43.63%**	**67.73%**	**40.87%**	**64.01%**	**48.46%**	**71.44%**	4.28	282

Table 1. Classification results on simulated data. We compare the proposed joint architecture to classifiers that ingest (and are trained on) pre-processed images output by conventional ISPs, including Darktable and the Movidius Myriad 2 ISP, and learnable deep ISPs [Chen et al. 2018]. Off-the-shelf MobileNet-v1 classifiers pretrained on Imagenet perform poorly on ISP-preprocessed data (Movidius Myriad 2 ISP + Pretrained and Darktable ISP + Pretrained). Fine-tuning these classifiers on the ISP-processed data (Movidius Myriad 2 ISP + Finetuning, Movidius Myriad 2 ISP + Finetuning, and U-Net [Chen et al. 2018] deep ISP models) results in substantially improved performance. While the parameters of the conventional ISPs have been expert-tuned, we train the deep U-Net ISP from [Chen et al. 2018] on the clear/noisy training corpus, and we also report results when adding an perceptual loss [Johnson et al. 2016] (+ percep. loss). However, none of the fine-tuned models, trained on top of traditional or learnable ISPs, does outperform networks that *do not employ an ISP at all* across all settings, as evidenced by results of a MobileNet-v1 on unprocessed RAW data (From Scratch MobileNet-v1). Only the proposed joint architecture with a learned Anscombe network outperforms both, traditional pipelines, as well as from-scratch-trained models in both Top-1 and Top-5 classification accuracy across illumination conditions. The proposed approach even outperforms from-scratch-trained MobileNet-v2 models that are *deeper networks with larger network capacity* compared to our architecture. Note that all other models compared in this table use the MobileNet-v1 architecture as classifier heads. We highlight the best and second best models using bold and underlined text, respectively.

is prohibitively costly to implement in hardware. The parameters of the darktable RAW developing tool and the Movidius Myriad 2 were hand-tuned by a human expert to maximize perceptual quality. The results in Table 1, rows third and fourth, validate that the conventional approach of combining a high-quality ISP, optimized for human viewing, with an existing pretrained high-level network fails in low-light scenarios. *In all cases*, this approach performs weakly due to the severe noise present in the image data. This applies both to efficient hardware ISP architectures, such as the Movidius Myriad 2 ISP, as well as to high-performance photography RAW processing ISPs, such as Darktable. In fact, as detailed below, processing RAW measurements with conventional image processing units, tuned for perceptual quality, can decrease classification performance, compared to almost unprocessed bi-linearly interpolated color images. These findings also apply to image degradations introduced by optical aberrations. We refer the reader to the supplement for a study on the effect of optical aberrations.

Finetuning a classifier on top of ISP-preprocessed images does not do better than a model trained directly on RAW noisy data. Traditional ISP pipelines achieve acceptable performance only when the networks are fine-tuned, i.e. specialized, to the degraded low-light imaging data output by the respective ISP (fifth and sixth rows of Table 1). However, the performance of these specialized networks applied on the output of existing ISPs is exceeded by simply training a network from scratch for the given imaging condition but without an ISP at all, only using bilinearly demosaicked color images (first row of Table 1). The classifier trained without ISP preprocessing obtained higher Top-1 accuracy on three out of the four noise settings. Overall, processing images with conventional ISP pipelines, that are designed and tuned for human viewing, at best marginally increased classification accuracy for models specialized to individual light levels and in many cases substantially decreased performance. This is especially apparent for varying low-light conditions (columns 2-to-20 lux and 2-to-200 lux), where classifiers finetuned on ISP outputs obtain only half of the Top-1 accuracy of classifiers trained from scratch. On a first glance, this result may argue for completely removing traditional ISP pipelines and simply train standard CNN classifiers with as little traditional preprocessing as possible.

Anscombe Networks outperform classifiers trained from scratch on RAW data (even with larger model capacity). We compare the proposed method to MobileNet-v1 trained directly on RAW data and its deeper larger variant, MobileNet-v2; see first, second and last rows of Table 1. The MobileNet-v2 variant introduces inverted residual blocks, in which shortcut connections are introduced between bottleneck layers, and improves efficiency and accuracy relative to MobileNet-v1. Specifically, we use MobileNet-v2 (1.4) [Sandler et al. 2018], that has larger capacity (1.4 width multiplier) than the standard version. We observe that this deeper model improves results in almost all illumination conditions compared to MobileNet-v1. For the larger 2-to-200 lux illumination range, we do observe worse performance which we attribute to the larger architecture being slightly more prone to overfitting as a result of memorization. We note that our joint network obtains higher Top-1 and Top-5 accuracy compared to both models across all noise settings. Although MobileNet-v2 has a substantial higher number of parameters, this larger capacity does not translate into an improvement over our joint models. Finally, note that for both MobileNet networks, there is not explicit modeling of an intermediate image. These results validate that the improvement obtained by our joint architecture does not come from a larger capacity compared to the MobileNet-v1

version, but from the design of our Anscombe network. As such, we demonstrate that Anscombe Networks are highly effective at recovering an intermediate image representation that are tailored to the downstream task across different noise scenarios.

Anscombe Networks outperform classifiers trained on top of learnable deep ISP outputs. As a further comparison, we finetune a classifier network also on the outputs of existing learnable deep ISPs. Specfically, we train the deep ISP network from Chen et al. [2018] for image reconstruction with the loss settings proposed by the authors. We then finetune a MobileNet-v1 network on a corpus of denoised images. The results of this experiment are shown in the seventh row of Table 1, validating that our network also outperforms this approach for all noise scenarios. The margins are especially high for varying light levels (2-to-20 lux and 2-to-200 lux).

In addition, we also provide results using a perceptual loss [Johnson et al. 2016] in the first stage of the deep ISP training in addition to the ℓ_1-loss proposed in [Chen et al. 2018]. We manually optimized the weight ratios of both objective components. While this perceptual loss adds robustness across light level ranges, the proposed model maintains a high margin over this baseline. These results emphasize the efficiency and effectiveness of Anscombe Networks, which have 2.5× *and* 2× *fewer parameters and FLOPs*, respectively, compared to the learnable baselines.

Computational Complexity. The last two columns of Table 1 list the computational complexity of all models. For the deep ISP [Chen et al. 2018] with perceptual loss [Johnson et al. 2016], we do not consider the additional parameters of the pre-trained classifier used in the perceptual loss calculation, and for the Movidius and Darktable ISPs we only measure the MobileNet-v1 network compute cost, although modern ASIC ISPs require tremendous engineering efforts to be power efficient. For all models, we list complexity for RAW input images of 128x128 size. We note that the proposed joint architecture consists of *only* 0.1× *additional parameters (Anscombe network)* relative to MobileNet-v1, and this represents 2.5× fewer parameters than the deep ISP from [Chen et al. 2018]. Our joint architecture runs at 60fps.

Robustness and Generalization. The results in Table 1 validate the effectiveness of our proposed joint architecture to recover relevant information from RAW data to *improve accuracy for a high-level task*. We also emphasize our model's outstanding *generalization across different light levels*. We can see in Table 1 that while the accuracy of the models that use state-of-the-art software, hardware, and learnable ISPs drastically decrease over the 2-to-20 and 2-to-200 lux ranges, the accuracy of our proposed model remains stable. This again underlines the limitations of conventional models under more realistic scenarios, where light levels are highly variable, and the importance of building generalizable models. Our models also shines when comparing computational complexity, enabling robust real-time applications, as shown in Section 6.

Qualitative Interpretation. The results in Table 1 raise the question of why the jointly training Anscombe networks was so much more helpful to the classifier than conventional algorithms. The images in Fig. 5 suggest an answer. Fig. 5 shows a low-light image that was incorrectly classified by the pretrained MobileNet network but correctly classified by the joint architecture[1]. The RAW input image and a bilinearly demosaicked image is shown, as well as the outputs of the conventional hardware and software ISPs, and the intermediate mid-level representation produced by the Anscombe network unit. The label assigned by the classifier is given in each instance, as well as the PSNR and SSIM relative to the original image.

The images output by conventional ISPs for human viewing contain less noise than unprocessed RAW data. Fine details of the target class are blurred out, however. Comparing conventional and learned ISP outputs with Anscombe network's intermediates, we hypothesize that our joint Anscombe architecture tailors processing to the classification task by selectively boosting contrast around structures of the target class while removing noise in large smooth regions. This selective processing seems to be key to recover the target class structures independently of the noise or light level, which explains the robustness of our model across different light levels.

As a result, by conventional metrics of restoration quality such as SSIM and PSNR, the joint unit is, in fact, worse than conventional algorithms. These metrics do not distinguish between scene content necessary for a classification and background regions without task-specific information. We can also see, though that it preserves and amplifies detail that is useful to the classification network, the proposed Anscombe network does perform denoising and deblurring of the image. The qualitative results suggest traditional reconstruction algorithms and metrics used to make images visually pleasing to humans are not appropriate for high-level analytic tasks.

5.2 Evaluation in Low-light Imaging and Perception – Captured RAW Data

We demonstrate generalization of the proposed models to real-world low-light images. Using a Google Pixel phone rear camera, we collected low-light image patches in the wild. Rather than adopting the lengthy process of extracting these patches from objects at various scales in arbitrary photographs, we acquire full-frame images that directly correspond to classes in the ImageNet dataset and create patches by subsampling. While not affecting per-pixel noise, this process enables us to eliminate the effect of blur in the capture, allowing us to make solid claims about the effect of noise in isolation. The same applies to demosaicking which typically only considers a small neighborhood of pixels. We collect a low-light dataset approximately corresponding to light levels between 1 lux and 200 lux. The dataset consists of 1103 images across 40 imagenet classes respectively Table 2 lists results on the real-world dataset, including ablations of our proposed architecture. The evaluated models correspond to the 2 to 200 lux models in Table 1. These experiments evaluate the generalization performance of the respective models to real captured data. Absolute performance is worse than on the simulated datasets, which is likely due to a mismatch between how classes appear in ImageNet and how they appear in the wild. The relative margins are consistent with the simulated results.

[1]Please see Supplemental document for additional visualizations of the finetuned outputs of the deep ISP from [Chen et al. 2018].
[2]We do not count the parameters and FLOPS of the proprietary Pixel ISP here.

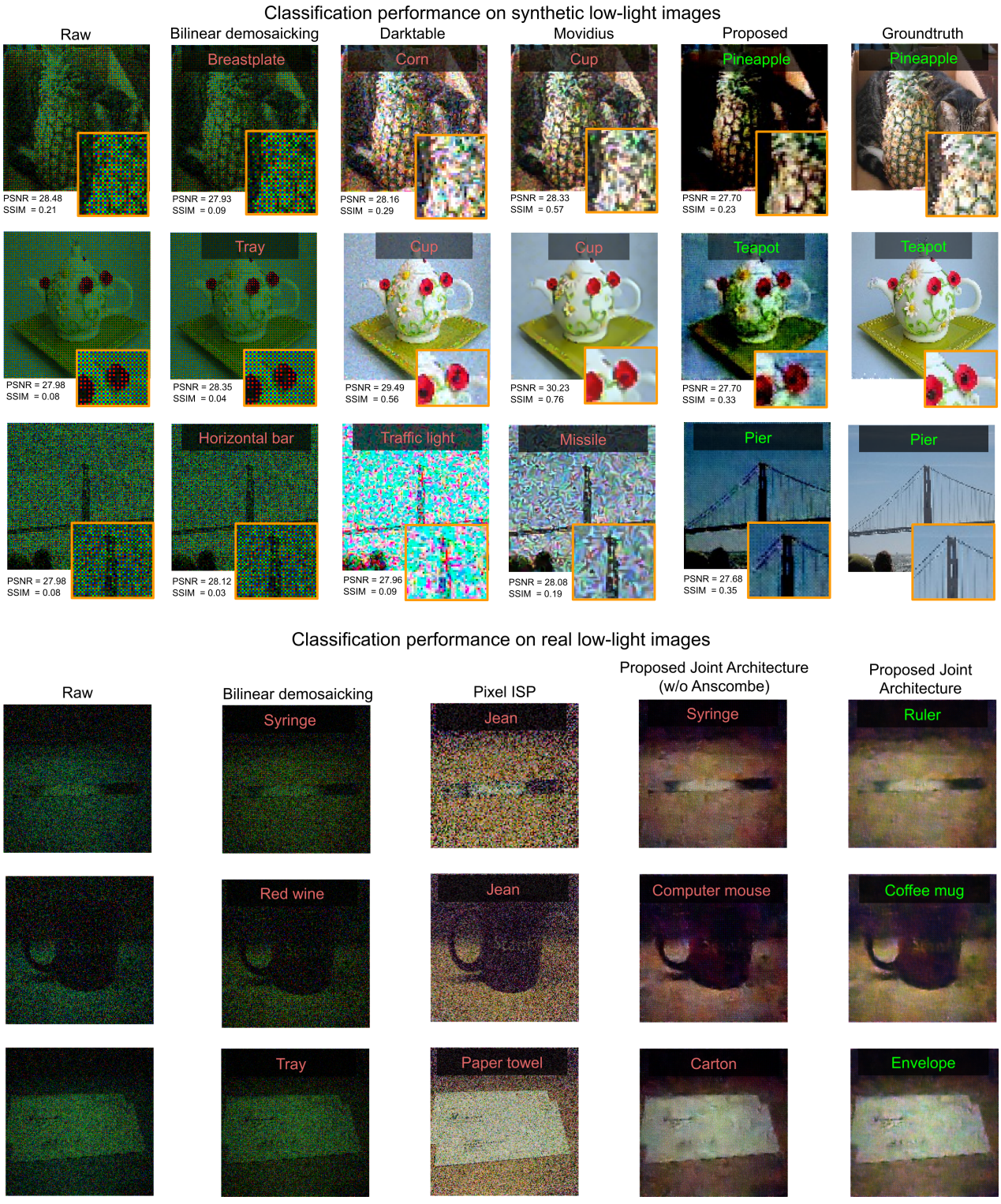

Fig. 5. Synthetic and real low-light RAW images, and their corresponding classification results after processing with conventional ISPs (Darktable and Movidius), bilinear demosaicking, and Anscombe Networks. The proposed joint architecture scores lower in terms of PSNR and SSIM. However, results suggest that our proposed model does not only removes noise, but selectively amplifies structures of the target class, which seems to benefit the overall classification accuracy of the model.

	Top-1	Top-5	#Parameters	FLOPS
From Scratch MobileNet-v1	27.03%	52.45%	4.23	181
From Scratch MobileNet-v2	26.92%	<u>56.45%</u>	6.90	320
Pixel ISP [2] + Pretrained MobileNet-v1	1.4%	14.1%	4.23	181
U-Net + Anscombe layers + MobileNet-v1	<u>28.80%</u>	55.20%	11.99	537
Proposed Joint Architecture (no Anscombe layers)	28.53%	54.25%	4.28	282
Proposed Joint Architecture	**33.13%**	**58.36%**	4.28	282

Table 2. Results on data captured in the wild with a Google Pixel phone rear camera for models trained on 2 to 200 lux. The exposure time was fixed at 1/10000 and the ISO at 8000. Additional digital gain was applied to normalize brightness. The best and second best methods are highlighted in bold and underlined text, respectively.

Anscombe networks generalize well to real data. Table 2 confirms that the highest classification accuracy was achieved by the proposed joint model, with Top-1 and Top-5 accuracy up to 6% higher than the fine-tuned models. The from-scratch tuned MobileNet-v1 and MobileNet-v2 models outperform the pretrained MobileNet network on Pixel ISP substantially, by 10s of percent.

Ablations of the proximal operator and Anscombe layers. Table 2 also includes results of the joint architecture without the proximal operator network (fourth row) and without the Anscombe transform (fifth row). For additional comparison, the network without the proximal operator uses the larger U-Net architecture described by [Chen et al. 2018] while keeping the Anscombe transform at the input and its inverse at the output of this network. These experiments validate that the architecture that uses Anscombe transform outperfoms the one that does not include this transform by around 5% in both Top-1 and Top-5 accuracy. Also, replacing the proximal operator network with U-Net reduces the accuracy of our proposed model by 4% and 3% in Top-1 and Top-5 accuracy, respectively. This margin validates that the Anscombe network as a whole is key for the performance of the overall joint model including the high-level classification model.

Qualitative Results. Fig. 5 helps to explain the improvement in classification accuracy of the proposed joint model as compared to conventional+fine-tuned and from-scratch baselines. We show image examples that each went through four different classification pipelines: one without any processing except for bilinear demosaicking for viewing, one processed with conventional ISPs before the MobileNet network, and two other processed using jointly trained models with and without Anscombe layer. As with the simulated data, conventional ISPs produce visually pleasing images by removing severe noise to a certain extent. However, fine details are lost in the process, leading to an incorrect classification result. The proposed joint stack does preserve and amplifies fine detail necessary for correct classification of the images.

5.3 Single-Image RAW Image Reconstruction in Low-Light for Human Viewing

Next, we evaluate the proposed Anscombe network architecture when trained as an ISP replacement for human viewing. Specifically, we demonstrate joint demosaicking, denoising and tonemapping for human viewing on a single capture in low light, using the training and validation data set from [Chen et al. 2018]. We employ the identical Anscombe network architecture from Sec. 4 but, instead of concatenating this model with a higher-level domain-specific network, we minimize a loss $\mathcal{L}$ formulated directly on the output image of the Anscombe network. This loss penalizes the difference between the prediction for a noisy observation and the corresponding clean long-exposure capture processed by a conventional ISP (with settings for normal lighting conditions). We use an ℓ_1-loss after evaluating other alternative loss functions.

	PSNR	SSIM
Darktable ISP [3]	8.94	0.03
Chen et al. [2018]	28.88	0.79
Proposed	**29.14**	**0.81**

Table 3. Anscombe Networks for Single-Image Low-Light Photography (w/o Classification). PSNR and SSIM comparison for Darktable ISP, Chen et al. [2018]'s learned U-Net ISP, and the proposed method, using the same training and test dataset proposed by [Chen et al. 2018].

Anscombe Networks also achieve state-of-the-art low-light performance for human viewing. The results in Table 3 show that our proposed model also obtains state-of-the-art performance for low-light image processing for human viewing. Our method outperforms the U-Net-based deep ISP [Chen et al. 2018] qualitatively and quantitatively. We visualize RAW imaging results obtained by the evaluated methods in Fig. 6. In the presented low-light scenario, conventional ISPs fail due to the significant noise degradations affecting the RAW sensor readings. In particular, the darktable ISP produces severe chromatic artifacts in smooth image regions. Furthermore, fine details at object boundaries are severely distorted as a result of an edge-preserving denoising block. In contrast, the plain U-Net model proposed in [Chen et al. 2018] produces visually pleasing images without chromatic artifact and free of residual noise. Chen et al.'s method also over-smooths image regions, i.e. noise is suppressed at the cost of texture loss. This behavior is particularly prevalent in areas with high intensity variations, around depth and illumination edges. The proposed Anscombe network model is tailored to intensity-dependent noise, and hence restores fine detail without over-smoothing or re-introducing residual noise. The results validate that Anscombe networks have the potential to be not only a domain-specific replacement for conventional general-purpose ISPs when considering non-traditional perception tasks but also when specialized to processing images for human viewing. We note that *specialized domain-specific processing is the goal of the proposed approach.* We do not dismiss traditional ISPs when the downstream application is unknown but highlight the potential of domain-specific camera processing pipelines.

6 MOBILE PROTOTYPE

We have implemented our joint low/high-level classification architecture on a mobile smartphone prototype along with a remote TensorFlow model server. The smartphone front-end application

[3] Traditional processing pipelines suffer also from severe color and white-balance artifacts in low-light such that quantiative results offer little insight. See Figure 6 for qualitative examples.

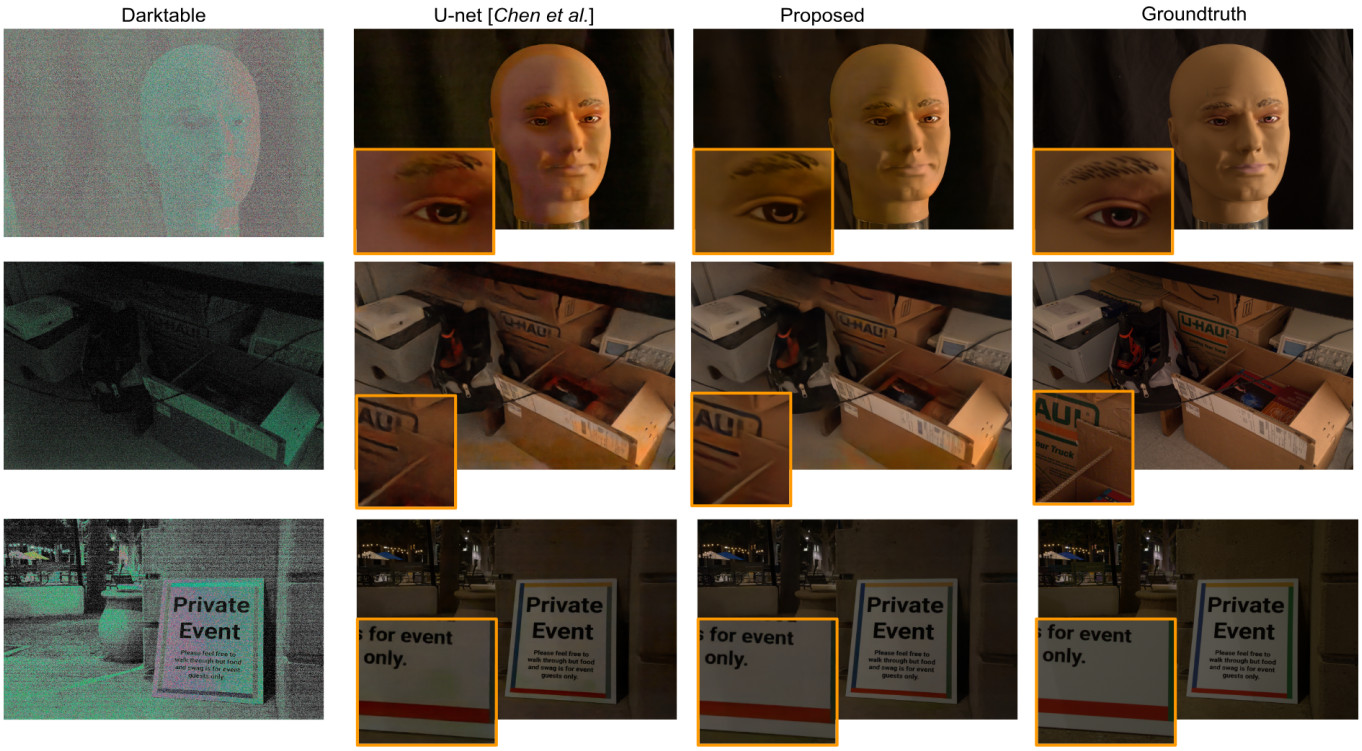

Fig. 6. Anscombe Networks for Single-Image Low-Light Photography (w/o Classification). Qualitative low-light denoising results for human viewing using the traditional Darktable ISP, the U-Net model proposed by [Chen et al. 2018], and the proposed Anscombe network. The proposed model and Chen et al. [2018] have been trained on the same low-light dataset from the dataset proposed in [Chen et al. 2018]. All methods use the same single RAW image as input.

handles all dynamic camera control and the capture itself. While we rely on the hardware ISP for control of white-balance and auto-focus, we manually fix the exposure to ensure repeatable measurements with consistent signal-to-noise ratio. We use the Android Camera2 API for capture control and acquisition of the raw measurements. The captured raw date is transferred to a remote instance using TensorFlow's high-performance protocol buffer serving system, which then performs the inference on the transferred data. We use an Amazon Web Services P2.1x GPU instance to host the servables for our joint models, and baseline models for comparison. Fig. 7 shows a photograph of the deployed application that classifies captures in the wild.

We achieve an inference throughput of about 60 FPS, while the vanilla MobileNet network performs at 80 FPS under the same conditions. Note that this performance is achieved without any inference optimization or integer-quantization, which frameworks such as TensorRT offer. We leave an efficient embedded hardware implementation as future work and note that variants of the MobileNet architecture already achieve interactive framerates on mobile devices [Howard et al. 2017].

6.0.1 Ultra Low-light Classification. The mobile prototype performs classification tasks robustly even in extreme low-light scenarios. Fig. 7 shows two such challenging capture scenarios along with classification results of the proposed and fine-tuned conventional MobileNet model. Both scenes were captured in a closed room without windows or other sources of ambient illumination. The only light sources present at the capture were the phone screen's illumination and the photocopier's dim LCD screen light. The scene captures shown in the left row of Fig. 7 were captured with a Canon Rebel T4 (f/4.5) with a long 2 second exposure at f/4.5. Note that even these DSLR setup shots are severely degraded by noise due to the low scene illumination. We acquired cellphone images with a long exposure of 125ms which, however, still allows for interactive frame-rates. The mobile prototype correctly performs classification even in these extreme imaging scenarios, where the from-scratch and fine-tuned MobileNet models fail. Please see the supplemental video for additional low-light classification results in the wild.

7 DISCUSSION

In summary, we showed that the performance of conventional imaging and perception stacks, combining a high-quality ISP for human viewing with high-level networks trained on clean JPEG datasets, fails in low-light capture scenarios (and with optical off-axis aberrations of inexpensive mobile optics). Moreover, training classification architectures from scratch without any ISP outperforms sequential fine-tuned architectures that include an ISP, seemingly advocating for the removal of an ISP for higher-level image analysis tasks.

In this work, we investigated learned processing architectures that perform end-to-end image processing and perception jointly.

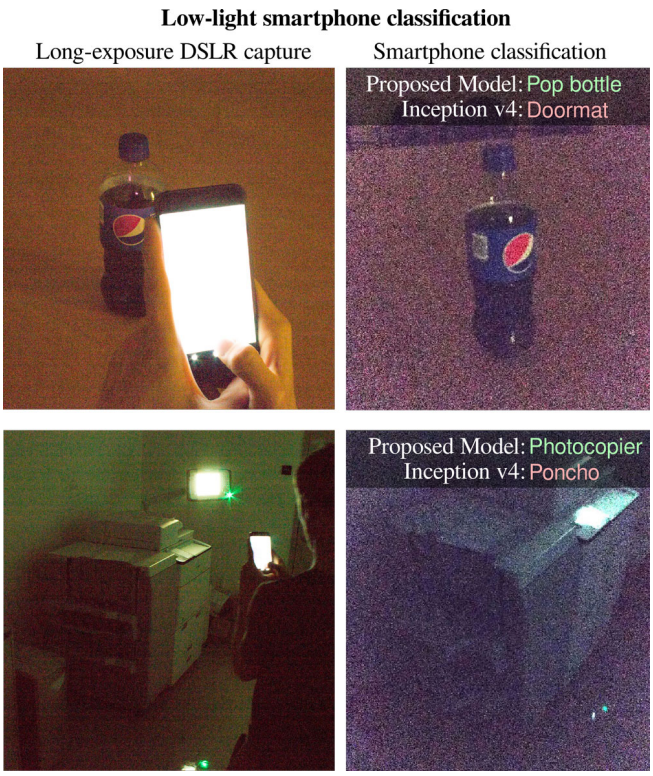

Fig. 7. Extreme low-light cellphone classification. Two scenes acquired without any light sources other than the cellphone screen (and printer LCD screen). The left column shows scene captures acquired over a long 2 second exposure using a Canon Rebel T4 DSLR camera. Note that these are still severely degraded by noise. The right column shows the corresponding mobile capture, acquired over a 125ms exposure, along with the classification label under these extreme conditions.

The proposed Anscombe networks act as an ISP, using RAW color filter array data as input, and is flexible to transfer to different sensor architectures and capture settings without retraining or capture of new training datasets. However, by making the model end-to-end differentiable, the architecture can be trained jointly for a high-level loss function, achieving state-of-the-art performance both for RAW image processing for human viewing and perception tasks across light levels from ultra-low light to well-lit scenes.

We demonstrated that the proposed architecture makes imaging and perception robust to the extreme capture scenarios that can be commonly found in real-world imaging. We highlighted major qualitative differences between sequential approaches and our joint end-to-end approach by visualizing intermediate representations in the proposed architecture and the output of conventional pipelines algorithms. We demonstrated that Anscombe networks generalize across camera architectures, including different CFA patterns, optical systems and noise models, promising that analogue neural ISPs can be developed for other sensors modalities across computational imaging, such as time-of-flight cameras, multi-spectral cameras, and sensor fusion systems.

Limitations and Future Work. While our proposed end-to-end model handles all the processing and image analysis after a RAW measurement has been acquired, a limitation of the method is that it does not address the dynamic control aspect of the capturing process, which is handled by the remaining trunk of the traditional ISP. Our proposed model then does not perform camera-control tasks, such as white-balance or auto-exposure. In the future, we plan to include auto-exposure and white-balance control in the proposed end-to-end model. These control tasks are particularly suited to include as image analysis feedback could severely affect the performance of these highly ill-posed problems.

In the future, we will also expand the proposed architecture to model the camera optics and sensors as unknowns. Just as we optimized the full perception and imaging stack, we aim to optimize the optics, CFA pattern, and other elements of the imaging system for the given high-level vision task, effectively learning not only the processing but also the camera architecture itself.

8 CONCLUSION

In the future a large portion of the images taken by cameras and other imaging systems will be consumed by high-level perception stacks, not by humans. We must reexamine the foundational assumptions of image processing in light of this momentous change. Image reconstruction algorithms designed to produce visually pleasing images for humans are not necessarily appropriate for a given perception task. We have proposed one approach to redesigning low-level processing pipelines in an end-to-end optimization framework, in a way that incorporates and benefits from knowledge of the physical image formation model and producing high-quality perceptually pleasing images when optimized for human-viewing.

ACKNOWLEDGMENTS

Vincent Sitzmann was supported by a Stanford Graduate Fellowship in Science and Engineering. Gordon Wetzstein was supported by a National Science Foundation (NSF) CAREER award (IIS 1553333), a Sloan Fellowship, a PECASE from the ARO, and by the KAUST Office of Sponsored Research through the Visual Computing Center CCF grant. Felix Heide was supported by an NSF CAREER Award (2047359).

REFERENCES

F. Agostinelli, M. Anderson, and H. Lee. 2013. Adaptive Multi-Column Deep Neural Networks with Application to Robust Image Denoising. In *Advances in Neural Information Processing Systems*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger (Eds.). 1493–1501.

A. Beck and M. Teboulle. 2009a. Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems. *IEEE Trans. Image Processing* 18, 11 (2009), 2419–2434.

A. Beck and M. Teboulle. 2009b. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences* 2, 1 (2009), 183–202.

Henryk Blasinski, Joyce Farrell, Trisha Lian, Zhenyi Liu, and Brian Wandell. 2018. Optimizing Image Acquisition Systems for Autonomous Driving. *Electronic Imaging* 2018, 5 (2018), 1–7.

T. Brooks, B. Mildenhall, T. Xue, J. Chen, D. Sharlet, and J. T. Barron. 2019. Unprocessing Images for Learned Raw Denoising. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11028–11037.

A. Buades, B. Coll, and J.-M. Morel. 2005. A non-local algorithm for image denoising. In *Proc. IEEE CVPR*, Vol. 2. 60–65.

Mark Buckler, Suren Jayasuriya, and Adrian Sampson. 2017. Reconfiguring the Imaging Pipeline for Computer Vision. In *IEEE International Conference on Computer Vision (ICCV)*. 975–984.

Antonin Chambolle and Thomas Pock. 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision* 40, 1 (2011), 120–145.

C. Chen, Q. Chen, J. Xu, and V. Koltun. 2018. Learning to See in the Dark. *ArXiv e-prints* (May 2018). arXiv:1805.01934

G. Chen, Y. Li, and S. Srihari. 2016. Joint visual denoising and classification using deep learning. In *Proceedings of the IEEE International Conference on Image Processing*. 3673–3677.

G. da Costa, W. Contato, T. Nazare, J. Neto, and M. Ponti. 2016. An empirical study on the effects of different types of noise in image classification tasks. *arXiv preprint arXiv:1609.02781* (2016).

K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. 2007. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Processing* 16, 8 (2007), 2080–2095.

darktable. 2018. darktable version 2.4.3. https://www.darktable.org.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 248–255.

S. Diamond, V. Sitzmann, F. Heide, and G. Wetzstein. 2017. Unrolled Optimization with Deep Priors. *arXiv preprint* (2017).

S. Dodge and L. Karam. 2016. Understanding how image quality affects deep neural networks. In *International Conference on Quality of Multimedia Experience*. 1–6.

A. Foi. 2009. Clipped noisy images: Heteroskedastic modeling and practical denoising. *Signal Processing* 89, 12 (2009), 2609–2629.

A. Foi and M. Makitalo. 2013. Optimal inversion of the generalized Anscombe transformation for Poisson-Gaussian noise. *IEEE Trans. Image Process.* 22, 1 (2013), 91–103.

A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian. 2008. Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Trans. Image Process.* 17, 10 (2008), 1737–1754.

Y. Ganin and V. Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*. 1180–1189.

M. Gharbi, G. Chaurasia, S. Paris, and F. Durand. 2016. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 191.

M. Gharbi, J. Chen, J. Barron, S. Hasinoff, and F. Durand. 2017. Deep Bilateral Learning for Real-Time Image Enhancement. *SIGGRAPH* (2017).

R. Glowinski and A. Marroco. 1975. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires. *Revue française d'automatique, informatique, recherche opérationnelle. Analyse numérique* 9, 2 (1975), 41–76.

S. Hasinoff, D. Sharlet, R. Geiss, A. Adams, J. Barron, F. Kainz, J. Chen, and M. Levoy. 2016. Burst Photography for High Dynamic Range and Low-light Imaging on Mobile Cameras. *ACM Trans. Graph.* 35, 6, Article 192 (2016), 12 pages.

James Hegarty, John Brunhaver, Zachary DeVito, Jonathan Ragan-Kelley, Noy Cohen, Steven Bell, Artem Vasilyev, Mark Horowitz, and Pat Hanrahan. 2014. Darkroom: Compiling High-level Image Processing Code into Hardware Pipelines. *ACM Trans. Graph. (SIGGRAPH)* 33, 4 (2014).

F. Heide, M. Steinberger, Y.-T. Tsai, M. Rouf, D. Pajak, D. Reddy, O. Gallo, J. Liu, W. Heidrich, K. Egiazarian, J. Kautz, and K. Pulli. 2014. FlexISP: A flexible camera image processing framework. *ACM Trans. Graph. (SIGGRAPH Asia)* 33, 6 (2014).

Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. 2017. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213* (2017).

Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *CoRR* abs/1704.04861 (2017). arXiv:1704.04861 http://arxiv.org/abs/1704.04861

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*. 448–456.

ISO 12233:2014 2014. ISO 12233:2014 Photography – Electronic still picture imaging – Resolution and spatial frequency responses.

A. Jalalvand, W. De Neve, R. Van de Walle, and J. Martens. 2016. Towards using Reservoir Computing Networks for noise-robust image recognition. In *Proceedings of the International Joint Conference on Neural Networks*. 1666–1672.

Ronnachai Jaroensri, Camille Biscarrat, Miika Aittala, and Frédo Durand. 2019. Generating Training Data for Denoising Real RGB Images via Camera Pipeline Simulation. *ArXiv* abs/1904.08825 (2019).

E. Jin, J. Phillips, S. Farnand, M. Belska, V. Tran, E. Chang, Y. Wang, and B. Tseng. 2017. Towards the Development of the IEEE P1858 CPIQ Standard–A validation study. *Electronic Imaging* 2017, 12 (2017), 88–94.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*. Springer, 694–711.

S. Karahan, M. Yildirum, K. Kirtac, F. Rende, G. Butun, and H. Ekenel. 2016. How Image Degradations Affect Deep CNN-Based Face Recognition?. In *International Conference of the Biometrics Special Interest Group*. 1–5.

Zhetong Liang, Jianrui Cai, Zisheng Cao, and Lei Zhang. 2019. CameraNet: A Two-Stage Framework for Effective Camera ISP Learning. *CoRR* abs/1908.01481 (2019). arXiv:1908.01481 http://arxiv.org/abs/1908.01481

Orly Liba, Kiran Murthy, Yun-Ta Tsai, Tim Brooks, Tianfan Xue, Nikhil Karnad, Qiurui He, Jonathan T. Barron, Dillon Sharlet, Ryan Geiss, Samuel W. Hasinoff, Yael Pritch, and Marc Levoy. 2019. Handheld Mobile Photography in Very Low Light. *ACM Trans. Graph.* 38, 6, Article 164 (Nov. 2019), 16 pages. https://doi.org/10.1145/3355089.3356508

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

M. Long, Y. Cao, J. Wang, and M. Jordan. 2015. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*. 97–105.

D. Moloney, B. Barry, R. Richmond, F. Connor, C. Brick, and D. Donohoe. 2014. Myriad 2: Eye of the computational vision storm. In *Hot Chips 26 Symposium (HCS), 2014 IEEE*. IEEE, 1–18.

A. Mosleh, P. Green, E. Onzon, I. Begin, and J.M. Pierre Langlois. 2015. Camera Intrinsic Blur Kernel Estimation: A Reliable Framework. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

ON Semi MT9P111. 2015. MT9P111: 1/4-Inch 5 Mp System-On-A-Chip (SOC) CMOS Digital Image Sensor. http://www.onsemi.com/pub/Collateral/MT9P111-D.PDF.

N. Parikh and S. Boyd. 2013. Proximal algorithms. *Foundations and Trends in Optimization* 1, 3 (2013), 123–231.

Jonathan B Phillips and Henrik Eliasson. 2018. *Camera Image Quality Benchmarking*. John Wiley & Sons.

Tobias Plotz and Stefan Roth. 2017. Benchmarking Denoising Algorithms With Real Photographs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

R. Ramanath, W. Snyder, Y. Yoo, and M. Drew. 2005a. Color image processing pipeline in digital still cameras. *IEEE Signal Processing Magazine* 22, 1 (2005), 34–43.

Rajeev Ramanath, Wesley E Snyder, Youngjun Yoo, and Mark S Drew. 2005b. Color image processing pipeline. *IEEE Signal Processing Magazine* 22, 1 (2005), 34–43.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *CoRR* abs/1505.04597 (2015). arXiv:1505.04597 http://arxiv.org/abs/1505.04597

M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4510–4520.

E. Schwartz, R. Giryes, and A. M. Bronstein. 2019. DeepISP: Toward Learning an End-to-End Image Processing Pipeline. *IEEE Transactions on Image Processing* 28, 2 (2019), 912–923.

L. Shao, R. Yan, X. Li, and Y. Liu. 2014. From Heuristic Optimization to Dictionary Learning: A Review and Comprehensive Comparison of Image Denoising Algorithms. *IEEE Transactions on Cybernetics* 44, 7 (2014), 1001–1013.

R. Stead. 2016. P2020 - Standard for Automotive System Image Quality. https://standards.ieee.org/develop/project/2020.html.

B. Sun and K. Saenko. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops*. Springer, 443–450.

Y. Tang and C. Eliasmith. 2010. Deep networks for robust visual recognition. In *Proceedings of the International Conference on Machine Learning*. 1055–1062.

Y. Tang, R. Salakhutdinov, and G. Hinton. 2012. Robust Boltzmann machines for recognition and denoising. In *Proceedings of Computer Vision and Pattern Recognition*. 2264–2271.

Ethan Tseng, Felix Yu, Yuting Yang, Fahim Mannan, Karl ST Arnaud, Derek Nowrouzezahrai, Jean-François Lalonde, and Felix Heide. 2019. Hyperparameter optimization in black-box image processing using differentiable proxies. *ACM Transactions on Graphics (SIGGRAPH)* 38, 4 (2019), 27.

E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. 2017. Adversarial discriminative domain adaptation. *arXiv preprint arXiv:1702.05464* (2017).

E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474* (2014).

I. Vasiljevic, A. Chakrabarti, and G. Shakhnarovich. 2016. Examining the Impact of Blur on Recognition by Convolutional Networks. *arXiv preprint arXiv:1611.05760* (2016).

Lucie Yahiaoui, Ciarán Hughes, Jonathan Horgan, Brian Deegan, Patrick Denny, and Senthil Yogamani. 2019. Optimization of ISP parameters for object detection algorithms. *Electronic Imaging* 2019, 15 (2019), 44–1.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. 2016. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *arXiv preprint arXiv:1608.03981* (2016).

L. Zhang, X. Wu, A. Buades, and X. Li. 2011. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic Imaging* 20, 2 (2011), 023016.