Replacing Mobile Camera ISP with a Single Deep Learning Model

Andrey Ignatov
andrey@vision.ee.ethz.ch

Luc Van Gool
vangool@vision.ee.ethz.ch

Radu Timofte
timofter@vision.ee.ethz.ch

ETH Zurich, Switzerland

Abstract

As the popularity of mobile photography is growing constantly, lots of efforts are being invested now into building complex hand-crafted camera ISP solutions. In this work, we demonstrate that even the most sophisticated ISP pipelines can be replaced with a single end-to-end deep learning model trained without any prior knowledge about the sensor and optics used in a particular device. For this, we present PyNET, a novel pyramidal CNN architecture designed for fine-grained image restoration that implicitly learns to perform all ISP steps such as image demosaicing, denoising, white balancing, color and contrast correction, demoireing, etc. *The model is trained to convert RAW Bayer data obtained directly from mobile camera sensor into photos captured with a professional high-end DSLR camera, making the solution independent of any particular mobile ISP implementation. To validate the proposed approach on the real data, we collected a large-scale dataset consisting of 10 thousand full-resolution RAW–RGB image pairs captured in the wild with the Huawei P20 cameraphone (12.3 MP Sony Exmor IMX380 sensor) and Canon 5D Mark IV DSLR. The experiments demonstrate that the proposed solution can easily get to the level of the embedded P20's ISP pipeline that, unlike our approach, is combining the data from two (RGB + B/W) camera sensors. The dataset, pre-trained models and codes used in this paper are available on the project website:* https://people.ee.ethz.ch/~ihnatova/pynet.html

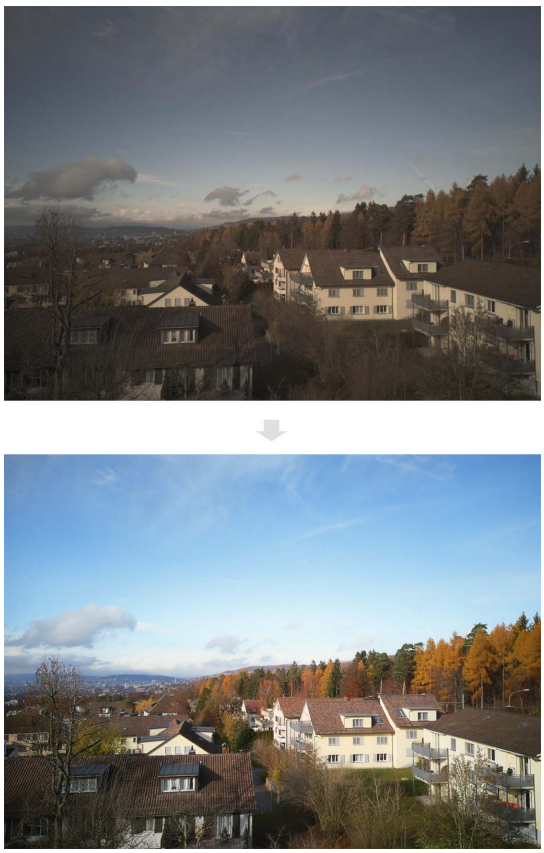

Figure 1. Huawei P20 RAW photo (visualized) and the corresponding image reconstructed with our method.

1. Introduction

While the first mass-market phones and PDAs with mobile cameras appeared in the early 2000s, at the beginning they were producing photos of very low quality, significantly falling behind even the simplest compact cameras. The resolution and quality of mobile photos have been growing constantly since that time, with a substantial boost after 2010, when mobile devices started to get powerful hardware suitable for heavy image signal processing (ISP) systems. Since then, the gap between the quality of photos from smartphones and dedicated point-and-shoot cameras is diminishing rapidly, and the latter ones have become nearly extinct over the past years. With this, smartphones became the main source of photos nowadays, and the role and requirements to their cameras have increased even more.

The modern mobile ISPs are quite complex software systems that are sequentially solving a number of low-level and global image processing tasks, such as image demosaicing, white balance and exposure correction, denoising

Figure 2. Typical artifacts appearing on photos from mobile cameras. From left to right: cartoonish blurring / "watercolor effect" (Xiaomi Mi 9, Samsung Galaxy Note10+), noise (iPhone 11 Pro, Google Pixel 4 XL) and image flattening (OnePlus 7 Pro, Huawei Mate 30 Pro).

and sharpening, color and gamma correction, *etc*. The parts of the system responsible for different subtasks are usually designed separately, taking into account the particularities of the corresponding sensor and optical system. Despite all the advances in the software stack, the hardware limitations of mobile cameras remain unchanged: small sensors and relatively compact lenses are causing the loss of details, high noise levels and mediocre color rendering. The current classical ISP systems are still unable to handle these issues completely, and are therefore trying to hide them either by flattening the resulting photos or by applying the "watercolor effect" that can be found on photos from many recent flagship devices (see Figure 2). Though deep learning models can potentially deal with these problems, and besides that can be also deployed on smartphones having dedicated NPUs and AI chips [24, 23], their current use in mobile ISPs is still limited to scene classification or light photo post-processing.

Unlike the classical approaches, in this paper we propose to learn the entire ISP pipeline with only one deep learning model. For this, we present an architecture that is trained to map RAW Bayer data from the camera sensor to the target high-quality RGB image, thus intrinsically incorporating all image manipulation steps needed for fine-grained photo restoration (see Figure 1). Since none of the existing mobile ISPs can produce the required high-quality photos, we are collecting the target RGB images with a professional Canon 5D Mark IV DSLR camera producing clear noise-free high-resolution pictures, and present a large-scale image dataset consisting of 10 thousand RAW (phone) / RGB (DSLR) photo pairs. As for mobile camera, we chose the Huawei P20 cameraphone featuring one of the most sophisticated mobile ISP systems at the time of the dataset collection.

Our main contributions are:

- An end-to-end deep learning solution for RAW-to-RGB image mapping problem that is incorporating all image signal processing steps by design.
- A novel PyNET CNN architecture designed to combine heavy global manipulations with low-level fine-grained image restoration.
- A large-scale dataset containing 10K RAW–RGB image pairs collected in the wild with the Huawei P20 smartphone and Canon 5D Mark IV DSLR camera.
- A comprehensive set of experiments evaluating the quantitative and perceptual quality of the reconstructed images, as well as comparing the results of the proposed deep learning approach with the results obtained with the built-in Huawei P20's ISP pipeline.

2. Related Work

While the problem of real-world RAW-to-RGB image mapping has not been addressed in the literature, a large number of works dealing with various image restoration and enhancement tasks were proposed during the past years.

Image super-resolution is one of the most classical image reconstruction problems, where the goal is to increase image resolution and sharpness. A large number of efficient solutions were proposed to deal with this task [1, 54], starting from the simplest CNN approaches [10, 27, 50] to complex GAN-based systems [31, 46, 59], deep residual models [34, 68, 55], Laplacian pyramid [30] and channel attention [67] networks. Image deblurring [6, 48, 38, 51] and denoising [65, 64, 66, 52] are the other two related tasks targeted at removing blur and noise from the pictures.

A separate group of tasks encompass various global image adjustment problems. In [62, 13], the authors proposed solutions for automatic global luminance and gamma adjustment, while work [5] presented a CNN-based method for image contrast enhancement. In [61, 32], deep learning solutions for image color and tone corrections were proposed, and in [47, 37] tone mapping algorithms for HDR images were presented.

The problem of comprehensive image quality enhancement was first addressed in [19, 20], where the authors proposed to enhance all aspects of low-quality smartphone photos by mapping them to superior-quality images obtained with a high-end reflex camera. The collected DPED dataset was later used in many subsequent works [41, 9, 57, 18, 35] that have significantly improved the results on this problem. Additionally, in [22] the authors examined the possibility of running the resulting image enhancement models directly

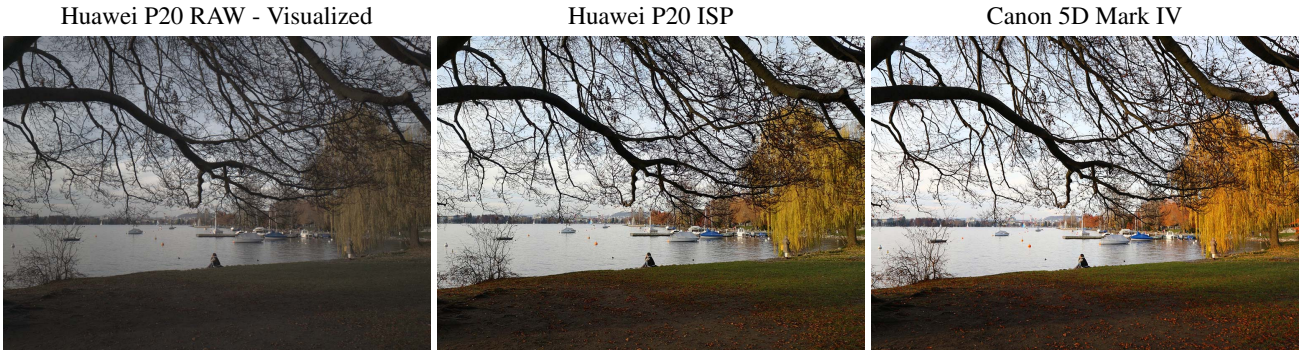

Figure 3. Example set of images from the collected Zurich RAW to RGB dataset. From left to right: original RAW image visualized with a simple ISP script, RGB image obtained with P20's built-in ISP system, and Canon 5D Mark IV target photo.

on smartphones, and proposed a number of efficient solutions for this task. It should be mentioned that though the proposed models were showing nice results, they were targeted at refining the images obtained with smartphone ISPs rather than processing RAW camera data.

While there exist many classical approaches for various image signal processing subtasks such as image demosaicing [33, 11, 15], denoising [3, 8, 12], white balancing [14, 56, 4], color correction [29, 43, 44], *etc*., only a few works explored the applicability of deep learning models to these problems. In [39, 53], the authors demonstrated that convolutional neural networks can be used for performing image demosaicing, and outperformed several conventional models in this task. Works [2, 16] used CNNs for correcting the white balance of RGB images, and in [63] deep learning models were applied to synthetic LCDMoire dataset for solving image demoireing problem. In [49], the authors collected 110 RAW low-lit images with Samsung S7 phone, and used a CNN model to remove noise and brighten demosaiced RGB images obtained with a simple hand-designed ISP. Finally, in work [42] RAW images were artificially generated from JPEG photos presented in [7], and a CNN was applied to reconstruct the original RGB pictures. In this paper, we will go beyond the constrained artificial settings used in the previous works, and will be solving *all* ISP subtasks on real data simultaneously, trying to outperform the commercial ISP system present in one of the best camera phones released in the past two years.

3. Zurich RAW to RGB dataset

To get real data for RAW to RGB mapping problem, a large-scale dataset consisting of 20 thousand photos was collected using Huawei P20 smartphone capturing RAW photos (plus the resulting RGB images obtained with Huawei's built-in ISP), and a professional high-end Canon 5D Mark IV camera with Canon EF 24mm f/1.4L fast lens. RAW data was read from P20's 12.3 MP Sony Exmor IMX380 Bayer camera sensor – though this phone has a second 20 MP monochrome camera, it is only used by Huawei's internal ISP system, and the corresponding images cannot be retrieved with any public camera API. The photos were captured in automatic mode, and default settings were used throughout the whole collection procedure. The data was collected over several weeks in a variety of places and in various illumination and weather conditions. An example set of captured images is shown in Figure 3.

Since the captured RAW–RGB image pairs are not perfectly aligned, we first performed their matching using the same procedure as in [19]. The images were first aligned globally using SIFT keypoints [36] and RANSAC algorithm [58]. Then, smaller patches of size 448×448 were extracted from the preliminary matched images using a non-overlapping sliding window. Two windows were moving in parallel along the two images from each RAW-RGB pair, and the position of the window on DSLR image was additionally adjusted with small shifts and rotations to maximize the cross-correlation between the observed patches. Patches with cross-correlation less than 0.9 were not included into the dataset to avoid large displacements. This procedure resulted in 48043 RAW-RGB image pairs (of size $448\times448\times1$ and $448\times448\times3$, respectively) that were later used for training / validation (46.8K) and testing (1.2K) the models. RAW image patches were additionally reshaped into the size of $224\times224\times4$, where the four channels correspond to the four colors of the RGBG Bayer filer. It should be mentioned that all alignment operations were performed only on RGB DSLR images, therefore RAW photos from Huawei P20 remained unmodified, containing the same values as were obtained from the camera sensor.

4. Proposed Method

The problem of RAW to RGB mapping is generally involving both global and local image modifications. The first ones are used to alter the image content and its high-level

properties, such as brightness, while balance or color rendition, while low-level processing is needed for tasks like texture enhancement, sharpening, noise removal, deblurring, *etc*. More importantly, there should be an interaction between global and local modifications, as, for example, content understanding is critical for tasks like texture processing or local color correction. While there exists many deep learning models targeted at one of these two problem types, their application to RAW to RGB mapping or to general image enhancement tasks is leading to the corresponding issues: VGG- [27], ResNet- [31] or DenseNet-based [17] networks cannot alter the image significantly, while models relying on U-Net [45] or Pix2Pix [25] architectures are not good at improving local image properties. To address this issue, in this paper we propose a novel PyNET CNN architecture that is processing image at different scales and combines the learned global and local features together.

4.1. PyNET CNN Architecture

Figure 4 illustrates schematic representation of the proposed deep learning architecture. The model has an inverted pyramidal shape and is processing the images at five different scales. The proposed architecture has a number of blocks that are processing feature maps in parallel with convolutional filters of different size (from 3×3 to 9×9), and the outputs of the corresponding convolutional layers are then concatenated, which allows the network to learn a more diverse set of features at each level. The outputs obtained at lower scales are upsampled with transposed convolutional layers, stacked with feature maps from the upper level and then subsequently processed in the following convolutional layers. *Leaky ReLU* activation function is applied after each convolutional operation, except for the output layers that are using *tanh* function to map the results to (-1, 1) interval. Instance normalization is used in all convolutional layers that are processing images at lower scales (levels 2-5). We are additionally using one transposed convolutional layer on top of the model that upsamples the images to their target size.

The model is trained sequentially, starting from the lowest layer. This allows to achieve good image reconstruction results at smaller scales that are working with images of very low resolution and performing mostly global image manipulations. After the bottom layer is trained, the same procedure is applied to the next level till the training is done on the original resolution. Since each higher level is getting upscaled high-quality features from the lower part of the model, it mainly learns to reconstruct the missing low-level details and refines the results. Note that the input layer is always the same and is getting images of size 224×224×4, though only a part of the training graph (all layers participating in producing the outputs at the corresponding scale) is trained.

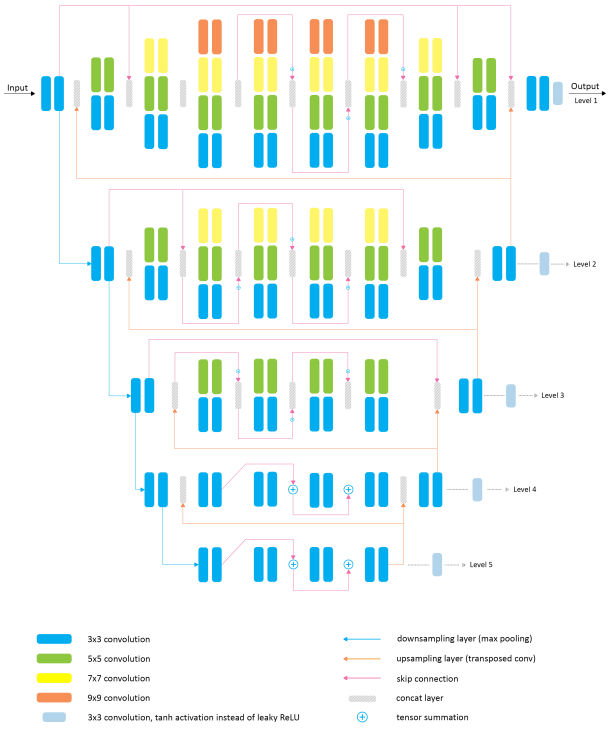

Figure 4. The architecture of the proposed PyNET model. Concat and Sum ops are applied to the outputs of the adjacent layers.

4.2. Loss functions

The loss function used to train the model depends on the corresponding level / scale of the produced images:

Levels 4-5 operate with images downscaled by a factor of 8 and 16, respectively, therefore they are mainly targeted at global color and brightness / gamma correction. These layers are trained to minimize the mean squared error (MSE) since the perceptual losses are not efficient at these scales.

Levels 2-3 are processing 2x / 4x downscaled images, and are mostly working on the global content domain. The goal of these layers is to refine the color / shape properties of various objects on the image, taking into account their semantic meaning. They are trained with a combination of the VGG-based [26] perceptual and MSE loss functions taken in the ratio of 4:1.

Level 1 is working on the original image scale and is primarily trained to perform local image corrections: texture enhancement, noise removal, local color processing, *etc*., while using the results obtained from the lower layers. It is trained using the following loss function:

$$\mathcal{L}_{\text{Level 1}} = \mathcal{L}_{\text{VGG}} + 0.75 \cdot \mathcal{L}_{\text{SSIM}} + 0.05 \cdot \mathcal{L}_{\text{MSE}},$$

where the value of each loss is normalized to 1. The structural similarity (SSIM) loss [60] is used here to increase the

Figure 5. Sample visual results obtained with the proposed deep learning method. Best zoomed on screen.

dynamic range of the reconstructed photos, while the MSE loss is added to prevent significant color deviations.

The above coefficients were chosen based on the results of the preliminary experiments on the considered RAW to RGB dataset. We should emphasize that each level is trained together with all (already pre-trained) lower levels to ensure a deeper connection between the layers.

4.3. Technical details

The model was implemented in TensorFlow [1] and was trained on a single *Nvidia Tesla V100* GPU with a batch size ranging from 10 to 50 depending on the training scale. The parameters of the model were optimized for $5 \sim 20$ epochs using ADAM [28] algorithm with a learning rate of $5e-5$. The entire PyNET model consists of 47.5M parameters, and it takes 3.8 seconds to process one 12MP photo (2944×3958 pixels) on the above mentioned GPU.

[1] https://github.com/aiff22/pynet

5. Experiments

In this section, we evaluate the quantitative and qualitative performance of the proposed solution on the real RAW to RGB mapping problem. In particular, our goal is to answer the following three questions:

- How well the proposed approach performs numerically and perceptually compared to common deep learning models widely used for various image-to-image mapping problems.
- How good is the quality of the reconstructed images in comparison to the built-in ISP system of the Huawei P20 camera phone.
- Is the proposed solution generalizable to other mobile phones / camera sensors.

To answer these questions, we trained a wide range of deep learning models including the SPADE [40],

Figure 6. Visual results obtained with 7 different architectures. From left to right, top to bottom: visualized RAW photo, SRCNN [10], VDSR [27], SRGAN [31], Pix2Pix [25], U-Net [45], DPED [19], our PyNET architecture, Huawei ISP image and the target Canon photo.

DPED [19], U-Net [45], Pix2Pix [25], SRGAN [31], VDSR [27] and SRCNN [10] on the same data and measured the obtained results. We performed a user study involving a large number of participants asked to rate the target DSLR photos, the photos obtained with P20's ISP pipeline and the images reconstructed with our method. Finally, we applied our pre-trained model to RAW photos from a different device – BlackBerry KeyOne smartphone, to see if the considered approach is able to reconstruct RGB images when using camera sensor data obtained with other hardware. The results of these experiments are described in detail in the following three sections.

5.1. Quantitative Evaluation

Before starting the comparison, we first trained the proposed PyNET model and performed a quick inspection of the produced visual results. An example of the reconstructed images obtained with the proposed model is shown in Figure 5. The produced RGB photos do not contain any notable artifacts or corruptions at both the local and global levels, and the only major issue is vignetting caused by camera optics. Compared to photos obtained with Huawei's ISP, the reconstructed images have brighter colors and more natural local texture, while their sharpness is slightly lower, which is visible when looking at zoomed-in images. We expect that this might be caused by P20's second 20 MP monochrome camera sensor that can be used for image sharpening. In general, the overall quality of photos obtained with Huawei's ISP and reconstructed with our method is quite comparable, though both of them are worse than the images produced by the Canon 5D DSLR in terms of the color and texture quality.

Next, we performed a quantitative evaluation of the proposed method and alternative deep learning approaches. Table 1 shows the resulting PSNR and MS-SSIM scores obtained with different deep learning architecture on the test subset of the considered RAW to RGB mapping dataset. All models were trained twice: with the original loss function and the one used for PyNET training, and the best result was selected in each case. As one can see, PyNET CNN was able to significantly outperform the other models in both the PSNR and MS-SSIM scores. The visual results obtained with these models (Figure 6) also confirm this conclusion. VGG-19 and SRCNN networks did not have enough power to perform good color reconstruction. The images produced by the SRGAN and U-Net architectures were too dark, with dull colors, while the Pix2Pix had significant problems with accurate color rendering – the results are looking unnaturally due to distorted tones. Considerably better image reconstruction was obtained with the DPED model, though in this case the images have a strong yellowish shade and are lacking the vividness. Unfortunately, the SPADE architecture cannot process images of arbitrary resolutions (the size of the input data should be the same as used during the training process), therefore we were unable to generate full images using this method.

Method	PSNR	MS-SSIM
PyNET	21.19	0.8620
SPADE [40]	20.96	0.8586
DPED [19]	20.67	0.8560
U-Net [45]	20.81	0.8545
Pix2Pix [25]	20.93	0.8532
SRGAN [31]	20.06	0.8501
VDSR [27]	19.78	0.8457
SRCNN [10]	18.56	0.8268

Table 1. Average PSNR/SSIM results on test images.

5.2. User Study

The ultimate goal of our work is to provide an alternative to the existing handcrafted ISPs and, starting from the camera's raw sensor readings, to produce DSLR-quality images

Figure 7. The results of the proposed method on RAW images from the BlackBerry KeyOne smartphone. From left to right: the original visualized RAW image, reconstructed RGB image and the same photo obtained with KeyOne's built-in ISP system using HDR mode.

for the end user of the smartphone. To measure the overall quality of our results, we designed a user study with the Amazon Mechanical Turk [2] platform.

For the user study we randomly picked test raw input images in full resolution to be processed by 3 ISPs (the basic Visualized RAW, Huawei P20 ISP, and PyNET). The subjects were asked to assess the quality of the images produced by each ISP solution in direct comparison with the reference images produced by the Canon 5D Mark IV DSLR camera. The rating scale for the image quality is as follows: 1 - 'much worse', 2 - 'worse', 3 - 'comparable', 4 - 'better', and 5 - 'much better' (image quality than the DSLR reference image). For each query comprised from an ISP result versus the corresponding DSLR image, we collected opinions from 20 different subjects. For statistical relevance we collected 5 thousand such opinions.

The Mean Opinion Scores (MOS) for each ISP approach are reported in Table 2. We note again that 3 is the MOS for image quality that is 'comparable' to the DSLR camera, while 2 corresponds to a clearly 'worse' quality. In this light, we conclude that the Visualized RAW ISP with a score of 2.01 is clearly 'worse' than the DSLR camera, while the ISP of the Huawei P20 camera phone gets 2.56, almost half way in between the 'worse' and 'comparable'. Our PyNET, on the other hand, with a score of 2.77 is substantially better than the innate ISP of the P20 camera phone, but also below the quality provided by the Canon 5D Mark IV DSLR camera.

RAW input	ISP	MOS ↑
Huawei P20	Visualized RAW	2.01
	Huawei P20 ISP	2.56
	PyNET (ours)	2.77
Canon 5D Mark IV		*3.00*

Table 2. Mean Opinion Scores (MOS) obtained in the user study for each ISP solution in comparison to the target DSLR camera (3 – comparable image quality, 2 – clearly worse quality).

In a direct comparison between the Huawei P20 ISP and our PyNET model (used now as a reference instead of the DSLR) with the same protocol and rating scale, we achieved a MOS of 2.92. This means that the P20's ISP produces images of poorer perceptual quality than our PyNET when starting from the same Huawei P20 raw images.

5.3. Generalization to Other Camera Sensors

While the proposed deep learning model was trained to map RAW images from a particular device model / camera sensor, we additionally tested it on a different smartphone to see if the learned manipulations can be transferred to other camera sensors and optics. For this, we have collected a number of images with the BlackBerry KeyOne smartphone

[2] https://www.mturk.com

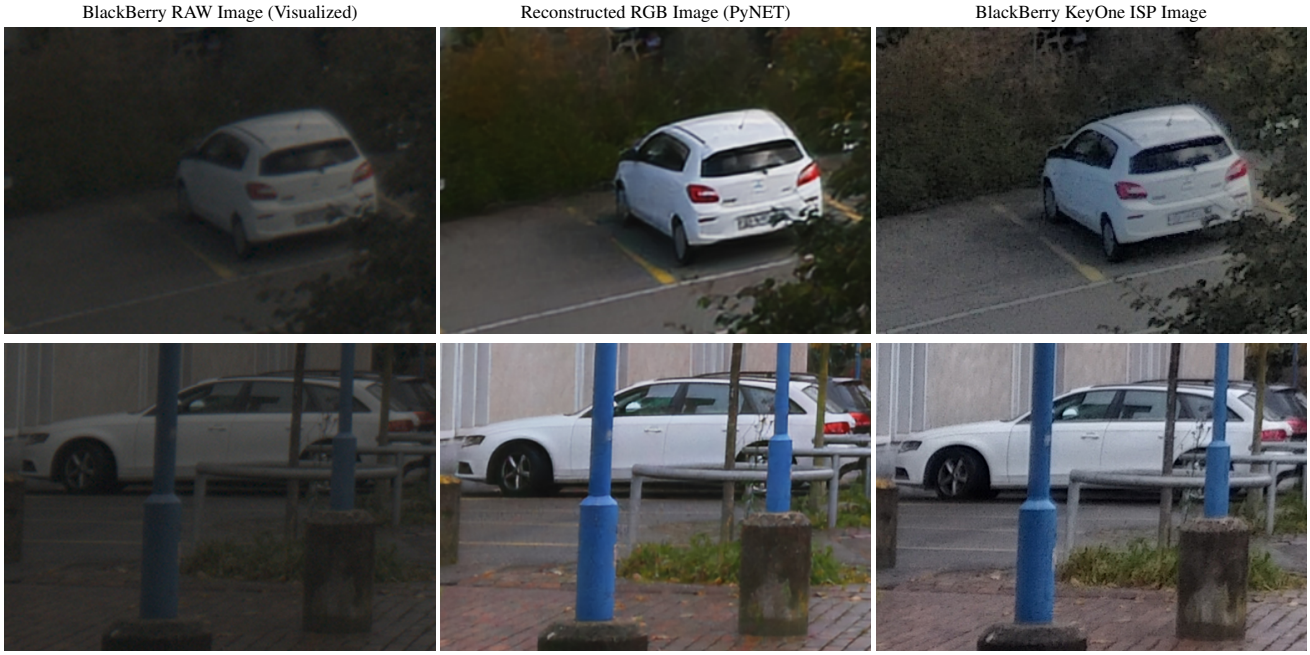

Figure 8. Image crops from the BlackBerry KeyOne RAW, reconstructed and ISP images, respectively.

that also has a 12 megapixel main camera, though is using a different sensor model (Sony IMX378) and a completely different optical system. RAW images were collected using the *Snap Camera HDR* [3] Android application, and we additionally shoot the same scenes with KeyOne's default camera app taking photos in HDR mode. The obtained RAW images were then fed to our pre-trained PyNET model, the resulting reconstruction results are illustrated in Figure 7.

As one can see, the PyNET model was able to reconstruct the image correctly and performed an accurate recovery of the colors, revealing many color shades not visible on the photos obtained with BlackBerry's ISP. While the latter images have a slightly higher level of details, PyNET has removed most of the noise present on the RAW photos as shown in Figure 8 demonstrating smaller image crops. Though the reconstructed photos are not ideal in terms of the exposure and sharpness, we should emphasize that the model was not trained on this particular camera sensor module, therefore much better results can be expected when tuning PyNET on the corresponding RAW–RGB dataset.

6. Conclusions

In this paper, we have investigated and proposed a change of paradigm – replacing an existing handcrafted ISP pipeline with a single deep learning model. For this, we first collected a large dataset of RAW images captured with the Huawei P20 camera phone and the corresponding paired RGB images from the Canon 5D Mark IV DSLR camera. Then, since the RAW to RGB mapping implies complex global and local non-linear transformations, we introduced PyNET, a versatile pyramidal CNN architecture. Next, we validated our PyNET model on the collected dataset and achieved significant quantitative PSNR and MS-SSIM improvements over the existing top CNN architectures. Finally, we conducted a user study to assess the perceptual quality of our ISP replacement approach. PyNET proved better perceptual quality than the handcrafted ISP innate to the P20 camera phone and closer quality to the target DSLR camera. We conclude that the results show the viability of our approach of an end-to-end single deep learned model as a replacement to the current handcrafted mobile camera ISPs. However, further study is required to fully grasp and emulate the flexibility of the current mobile ISP pipelines. We refer the reader to [21] for an application of PyNET to rendering natural camera bokeh effect and employing a new "Everything is Better with Bokeh!" dataset of paired wide and shallow depth-of-field images.

[3] https://play.google.com/store/apps/details?id=com.marginz.snaptrial

Acknowledgements

This work was partly supported by ETH Zurich General Fund (OK), by a Huawei project and by Amazon AWS and Nvidia grants.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, volume 3, page 2, 2017. 2

[2] Simone Bianco, Claudio Cusano, and Raimondo Schettini. Color constancy using cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 81–89, 2015. 3

[3] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005. 3

[4] Gershon Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 310(1):1–26, 1980. 3

[5] Jianrui Cai, Shuhang Gu, and Lei Zhang. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Transactions on Image Processing*, 27(4):2049–2062, 2018. 2

[6] Ayan Chakrabarti. A neural approach to blind motion deblurring. In *European conference on computer vision*, pages 221–235. Springer, 2016. 2

[7] Florian Ciurea and Brian Funt. A large image database for color constancy research. In *Color and Imaging Conference*, volume 2003, pages 160–164. Society for Imaging Science and Technology, 2003. 3

[8] Laurent Condat. A simple, fast and efficient approach to denoisaicking: Joint demosaicking and denoising. In *2010 IEEE International Conference on Image Processing*, pages 905–908. IEEE, 2010. 3

[9] Etienne de Stoutz, Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, and Luc Van Gool. Fast perceptual image enhancement. In *European Conference on Computer Vision Workshops*, 2018. 2

[10] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016. 2, 6

[11] Eric Dubois. Filter design for adaptive frequency-domain bayer demosaicking. In *2006 International Conference on Image Processing*, pages 2705–2708. IEEE, 2006. 3

[12] Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008. 3

[13] Xueyang Fu, Delu Zeng, Yue Huang, Yinghao Liao, Xinghao Ding, and John Paisley. A fusion-based enhancing method for weakly illuminated images. *Signal Processing*, 129:82–96, 2016. 2

[14] Arjan Gijsenij, Theo Gevers, and Joost Van De Weijer. Improving color constancy by photometric edge weighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):918–929, 2011. 3

[15] Keigo Hirakawa and Thomas W Parks. Adaptive homogeneity-directed demosaicing algorithm. *IEEE Transactions on Image Processing*, 14(3):360–369, 2005. 3

[16] Yuanming Hu, Baoyuan Wang, and Stephen Lin. Fc4: Fully convolutional color constancy with confidence-weighted pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4085–4094, 2017. 3

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 4

[18] Zheng Hui, Xiumei Wang, Lirui Deng, and Xinbo Gao. Perception-preserving convolutional networks for image enhancement on smartphones. In *European Conference on Computer Vision Workshops*, 2018. 2

[19] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. In *the IEEE Int. Conf. on Computer Vision (ICCV)*, 2017. 2, 3, 6

[20] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Wespe: weakly supervised photo enhancer for digital cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 691–700, 2018. 2

[21] Andrey Ignatov, Jagruti Patel, and Radu Timofte. Rendering natural camera bokeh effect with deep learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 8

[22] Andrey Ignatov and Radu Timofte. Ntire 2019 challenge on image enhancement: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[23] Andrey Ignatov, Radu Timofte, William Chou, Ke Wang, Max Wu, Tim Hartley, and Luc Van Gool. AI benchmark: Running deep neural networks on android smartphones. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2

[24] Andrey Ignatov, Radu Timofte, Andrei Kulik, Seungsoo Yang, Ke Wang, Felix Baum, Max Wu, Lirong Xu, and Luc Van Gool. Ai benchmark: All about deep learning on smartphones in 2019. *arXiv preprint arXiv:1910.06663*, 2019. 2

[25] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 4, 6

[26] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016. 4

[27] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016. 2, 4, 6

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[29] Ngai Ming Kwok, HY Shi, Quang Phuc Ha, Gu Fang, SY Chen, and Xiuping Jia. Simultaneous image color correction

and enhancement using particle swarm optimization. *Engineering Applications of Artificial Intelligence*, 26(10):2356–2371, 2013. 3

[30] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate superresolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, page 5, 2017. 2

[31] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, volume 2, page 4, 2017. 2, 4, 6

[32] Joon-Young Lee, Kalyan Sunkavalli, Zhe Lin, Xiaohui Shen, and In So Kweon. Automatic content-aware color and tone stylization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2470–2478, 2016. 2

[33] Xin Li, Bahadir Gunturk, and Lei Zhang. Image demosaicing: A systematic survey. In *Visual Communications and Image Processing 2008*, volume 6822, page 68221J. International Society for Optics and Photonics, 2008. 3

[34] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017. 2

[35] Hanwen Liu, Pablo Navarrete Michelini, and Dan Zhu. Deep networks for image to image translation with mux and demux layers. In *European Conference on Computer Vision Workshops*, 2018. 2

[36] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 3

[37] Kede Ma, Hojatollah Yeganeh, Kai Zeng, and Zhou Wang. High dynamic range image tone mapping by optimizing tone mapped image quality index. In *2014 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2014. 2

[38] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1628–1636, 2016. 2

[39] Bumjun Park and Jechang Jeong. Color filter array demosaicking using densely connected residual network. *IEEE Access*, 7:128076–128085, 2019. 3

[40] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2337–2346, 2019. 5, 6

[41] Zhu Pengfei, Huang Jie, Geng Mingrui, Ran Jiewen, Zhou Xingguang, Xing Chen, Wan Pengfei, and Ji Xiangyang. Range scaling global u-net for perceptual image enhancement on mobile devices. In *European Conference on Computer Vision Workshops*, 2018. 2

[42] Sivalogeswaran Ratnasingam. Deep camera: A fully convolutional neural network for image signal processing. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[43] Alessandro Rizzi, Carlo Gatta, and Daniele Marini. A new algorithm for unsupervised global and local color correction. *Pattern Recognition Letters*, 24(11):1663–1677, 2003. 3

[44] Alessandro Rizzi, Carlo Gatta, and Daniele Marini. From retinex to automatic color equalization: issues in developing a new algorithm for unsupervised color equalization. *Journal of Electronic Imaging*, 13(1):75–85, 2004. 3

[45] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4, 6

[46] Mehdi SM Sajjadi, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4491–4500, 2017. 2

[47] Yasir Salih, Aamir S Malik, Naufal Saad, et al. Tone mapping of hdr images: A review. In *2012 4th International Conference on Intelligent and Advanced Systems (ICIAS2012)*, volume 1, pages 368–373. IEEE, 2012. 2

[48] Christian J Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1439–1451, 2015. 2

[49] Eli Schwartz, Raja Giryes, and Alex M Bronstein. Deepisp: learning end-to-end image processing pipeline. *arXiv preprint arXiv:1801.06724*, 2018. 3

[50] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016. 2

[51] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 769–777, 2015. 2

[52] Pavel Svoboda, Michal Hradis, David Barina, and Pavel Zemcik. Compression artifacts removal using convolutional neural networks. *arXiv preprint arXiv:1605.00366*, 2016. 2

[53] Nai-Sheng Syu, Yu-Sheng Chen, and Yung-Yu Chuang. Learning deep convolutional networks for demosaicing. *arXiv preprint arXiv:1802.03769*, 2018. 3

[54] Radu Timofte, Shuhang Gu, Jiqing Wu, and Luc Van Gool. Ntire 2018 challenge on single image super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018. 2

[55] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4799–4807, 2017. 2

[56] Joost Van De Weijer, Theo Gevers, and Arjan Gijsenij. Edge-based color constancy. *IEEE Transactions on image processing*, 16(9):2207–2214, 2007. 3

[57] Thang Van Vu, Cao Van Nguyen, Trung X Pham, Tung Minh Liu, and Chang D. Youu. Fast and efficient image quality enhancement via desubpixel convolutional neural networks. In *European Conference on Computer Vision Workshops*, 2018. 2

[58] Andrea Vedaldi and Brian Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1469–1472. ACM, 2010. 3

[59] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *The European Conference on Computer Vision (ECCV) Workshops*, September 2018. 2

[60] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, volume 2, pages 1398–1402 Vol.2, Nov 2003. 4

[61] Zhicheng Yan, Hao Zhang, Baoyuan Wang, Sylvain Paris, and Yizhou Yu. Automatic photo adjustment using deep neural networks. *ACM Transactions on Graphics (TOG)*, 35(2):11, 2016. 2

[62] Lu Yuan and Jian Sun. Automatic exposure correction of consumer photographs. In *European Conference on Computer Vision*, pages 771–785. Springer, 2012. 2

[63] Shanxin Yuan, Radu Timofte, Gregory Slabaugh, and Ales Leonardis. Aim 2019 challenge on image demoireing: Dataset and study. *arXiv preprint arXiv:1911.02498*, 2019. 3

[64] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 2

[65] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938, 2017. 2

[66] Xin Zhang and Ruiyuan Wu. Fast depth image denoising and enhancement using a deep convolutional network. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2499–2503. IEEE, 2016. 2

[67] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 2

[68] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2472–2481, 2018. 2