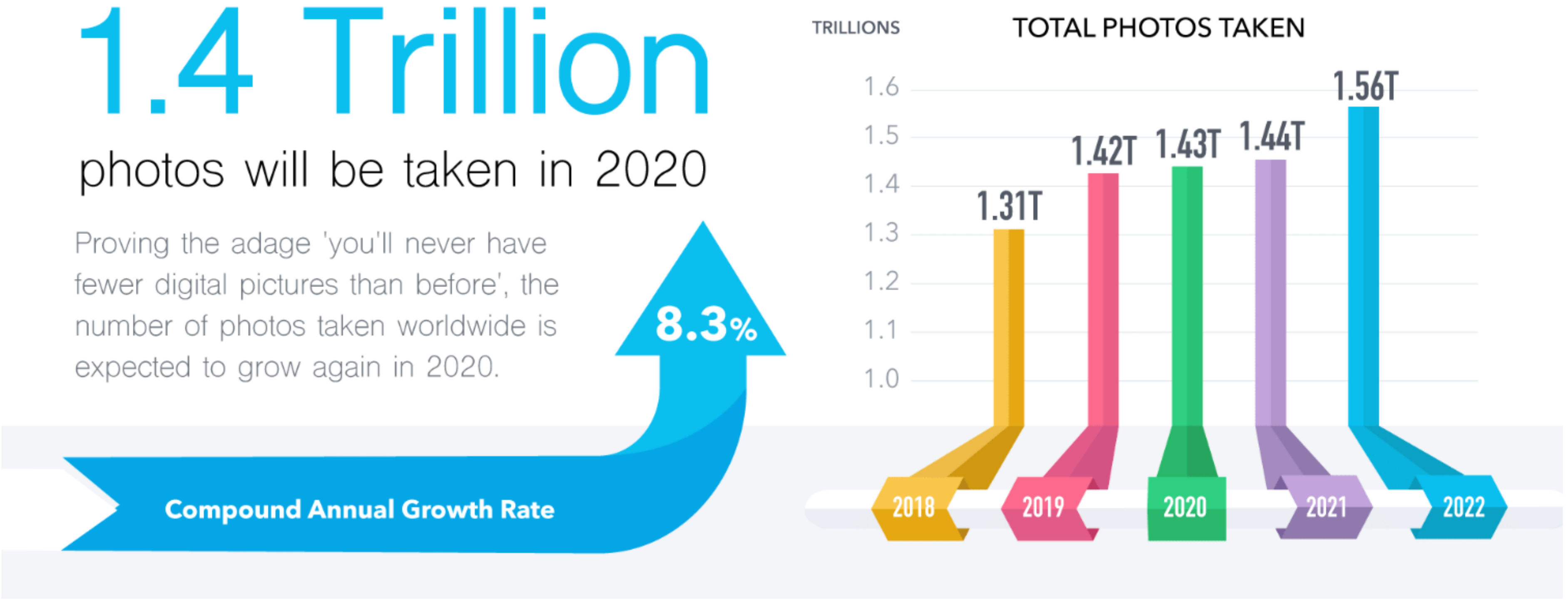
DeepLPF: Deep Local Parametric Filters for Image Enhancement

Sean Moran[1] , Pierre Marza[1,2], Steven McDonagh[1] , Sarah Parisot[1,3], Gregory Slabaugh[1]

Huawei Noah's Ark Lab[1] INSA Lyon[2] Mila Montréal[3]

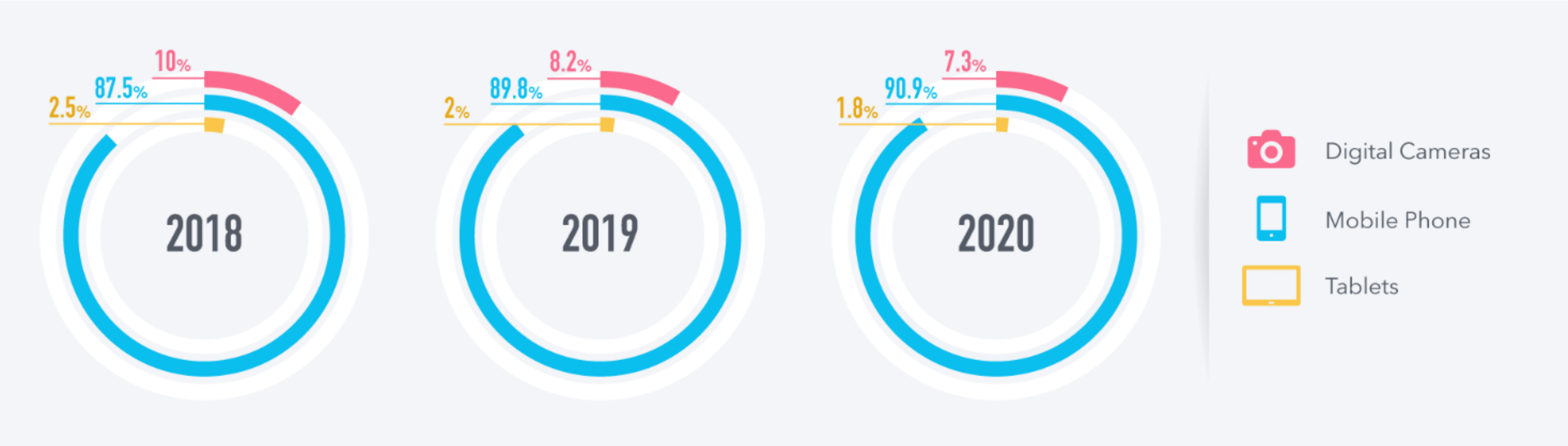
CVPR SEATTLE WASHINGTON JUNE 16-18 2020

We're taking a lot of photos…



https://focus.mylio.com/tech-today/how-many-photos-will-be-taken-in-2020

…on our smartphones



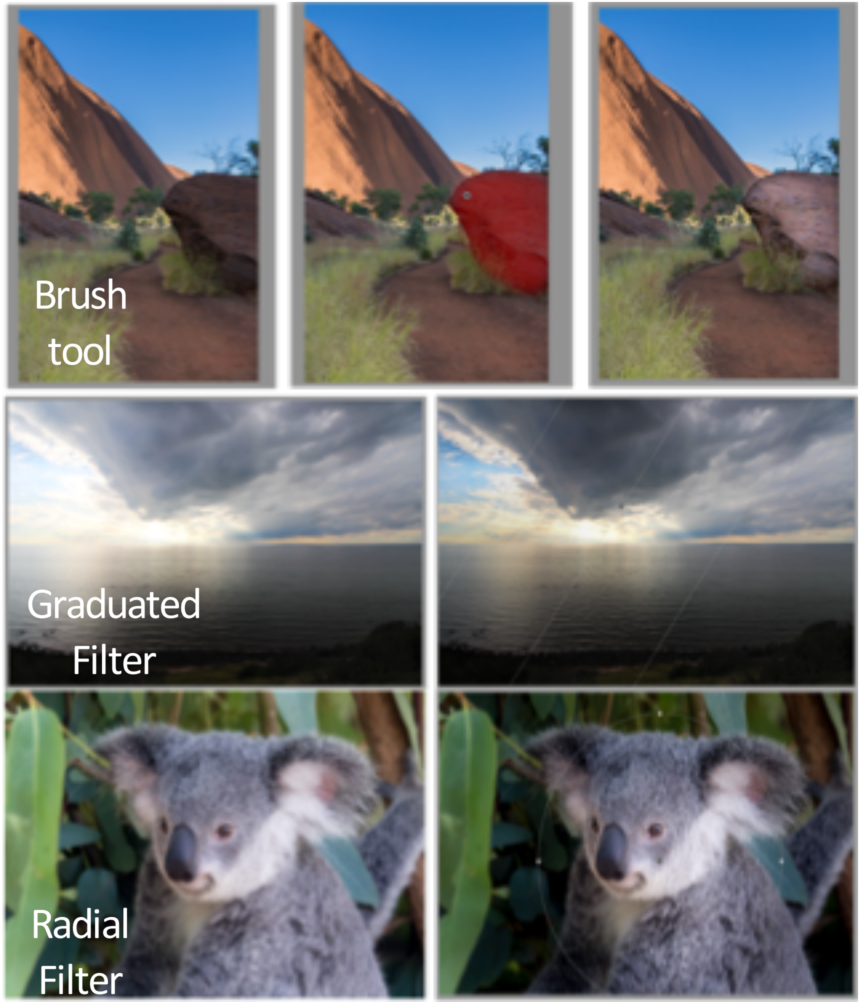
https://focus.mylio.com/tech-today/how-many-photos-will-be-taken-in-2020

But how to get more out of our photos?

https://www.clickinmoms.com/blog/reasons-deliberately-underexpose-photo/

Image retouching

- Image retouching requires **smooth local adjustments** as well as **global transformations**
- State of the art methods rely on either
  - **global** adjustments : **lacks fine-grained details**
  - **pixel-level** enhancements : **noisy** & **difficult to interpret**
- Professional artists typically use a combination of global and local enhancement tools to manually enhance images.
- Highly popular (e.g. in Lightroom, Photoshop) **local, parameterized enhancement tools** allow for smooth local adjustments



Radial filter

https://www.naturettl.com/master-the-graduated-radial-filters-in-lightroom/

https://creativepro.com/gradient-tools-lightroom/

Graduated filter

https://helpx.adobe.com/lightroom-classic/help/apply-local-adjustments.html

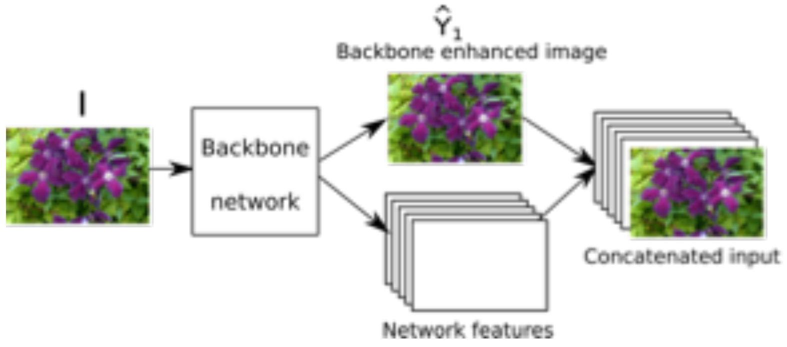
Brush tool

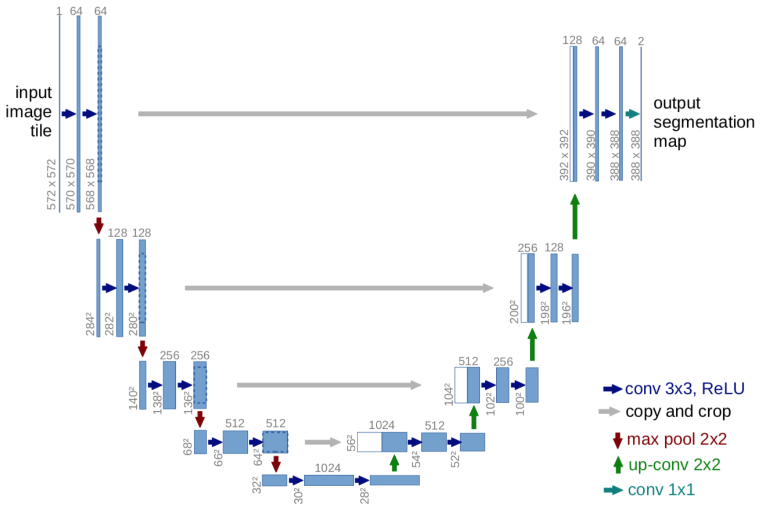
https://www.adobepress.com/articles/article.asp?p=1844834&seqNum=2

Deep Local Parametric Filters (DeepLPF)

- Introduction of **learnable** parametric **Elliptical**, **Graduated**, **Polynomial** image filters to reproduce artist local image retouching practices
- **Automatic** application to a photo
- **DeepLPF** : A novel architecture enabling **regression** of **spatially localized image filter parameters** for the target application of input image enhancement
- **Interpretable** & **intrinsically regularised** filters
- **Easier human feedback**
- A **plug-and-play** neural block with a filter fusion mechanism enabling the **learning of multiple filters simultaneously**

Method Overview

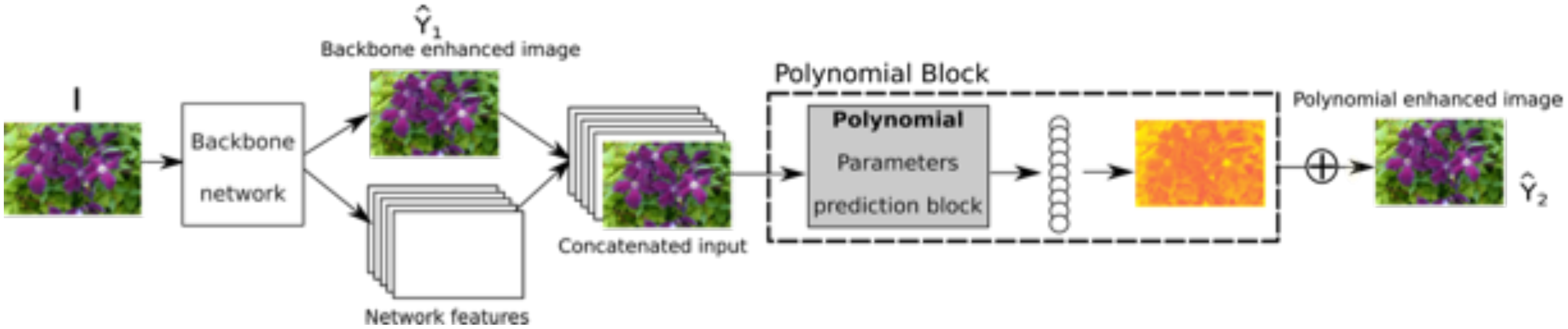




Global enhancement:

Standard backbone feature extractor (e.g. U-Net [1])

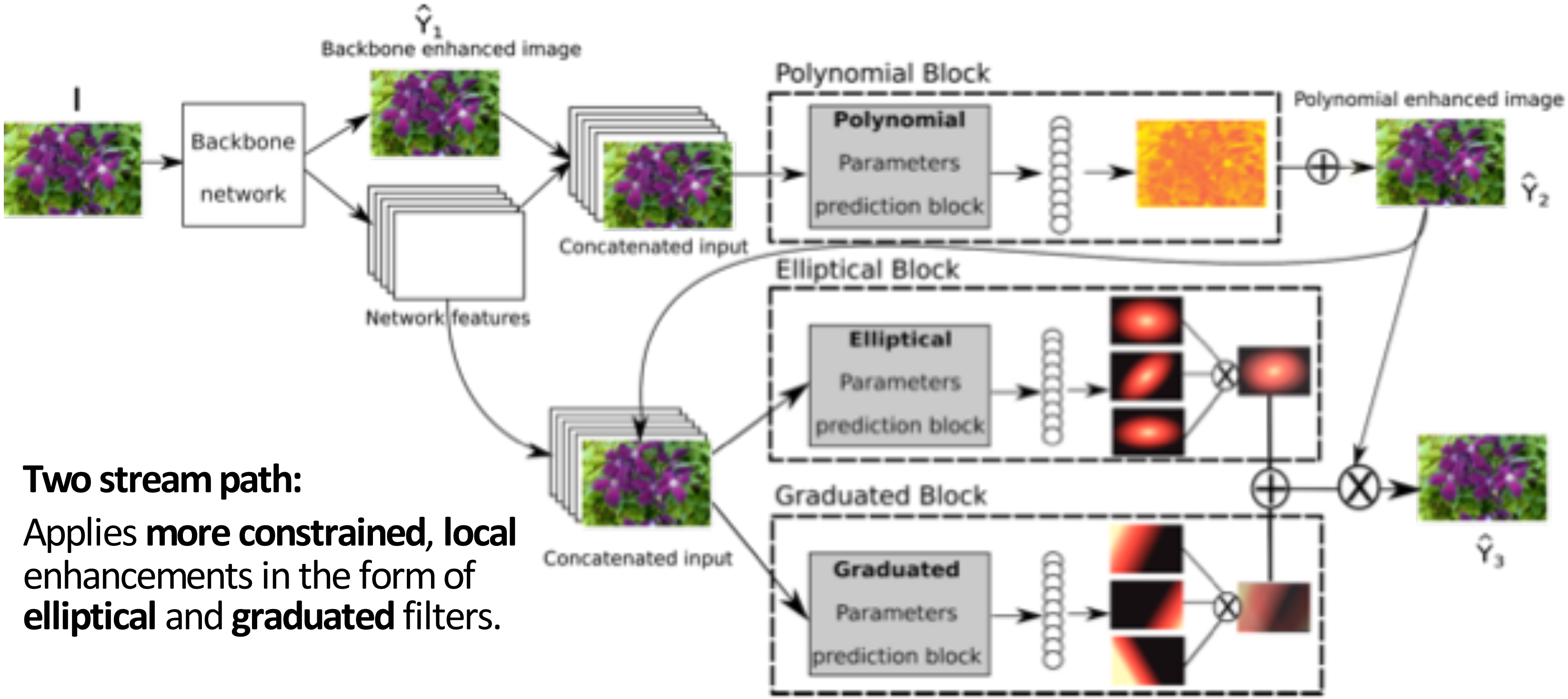
Method Overview



Single stream local enhancement path:

Estimates the parameters of a **polynomial filter** subsequently applied to the backbone enhanced image.

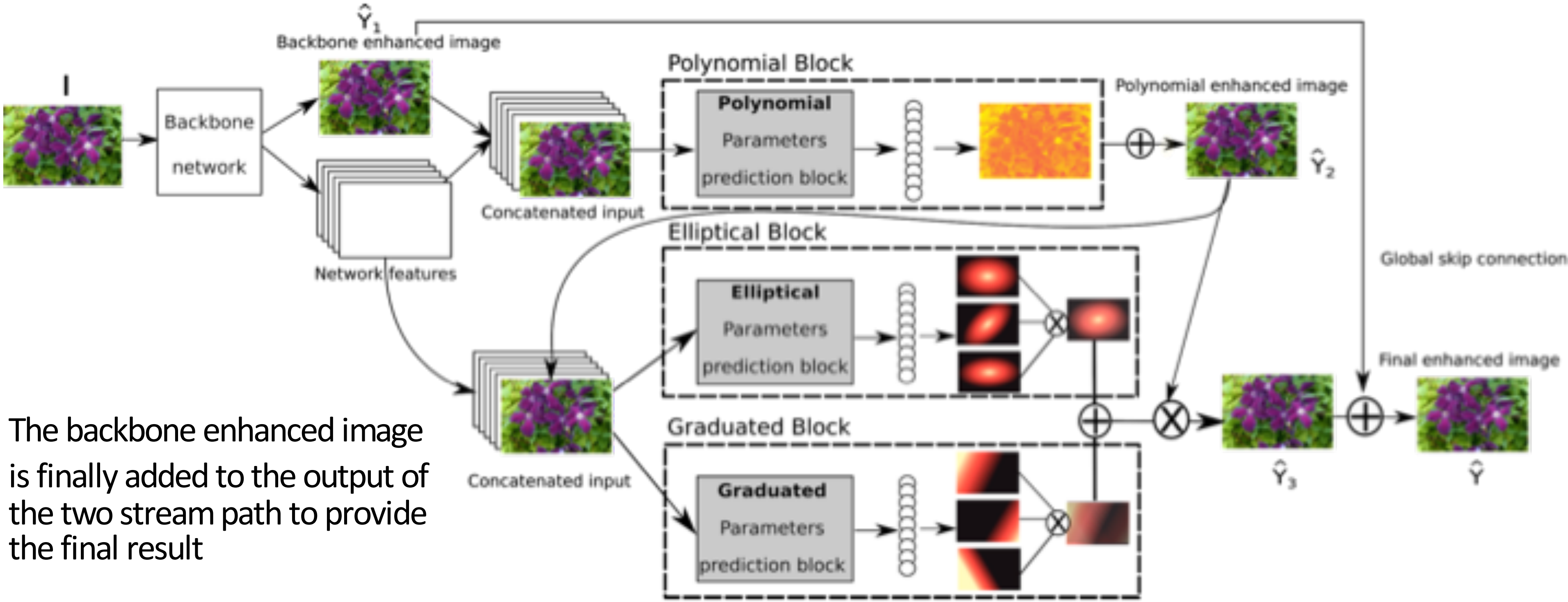
Method Overview



Two stream path:

Applies **more constrained**, **local** enhancements in the form of **elliptical** and **graduated** filters.

Method Overview

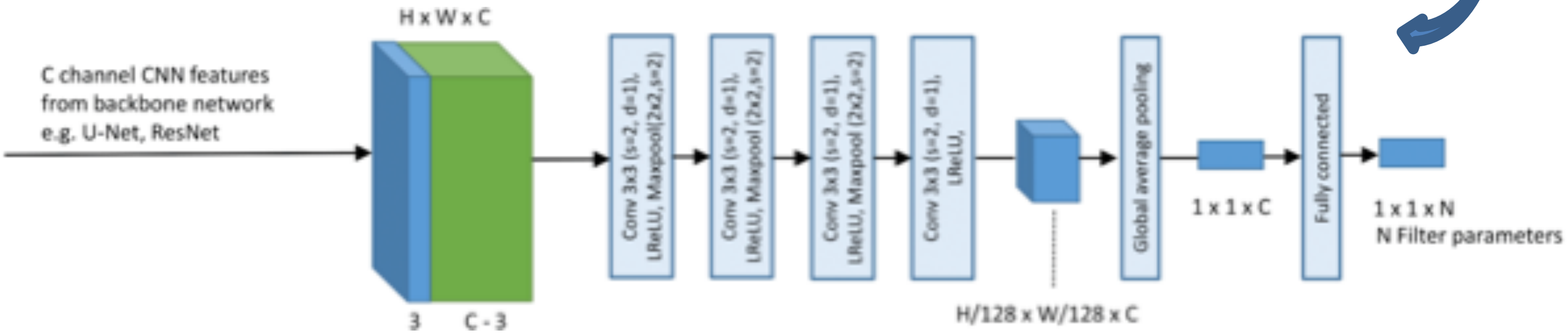


The backbone enhanced image is finally added to the output of the two stream path to provide the final result

Filter Parameter prediction block

- Each filter block (Polynomial, Graduated, Elliptical) **predicts the parameters of k instances of the corresponding filter type**

- k instances of a same filter type can be obtained by estimating $k * nb_{parameters}$ outputs.

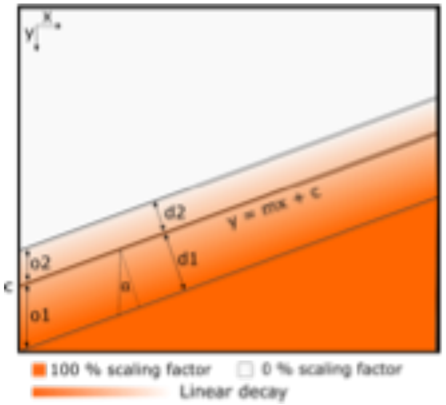


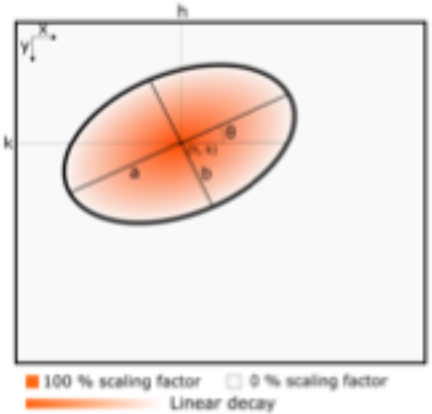
Filter Parametrisation

Graduated and Elliptical filters

- Geometrically regularized heatmaps
- Map each image pixel to **3 scaling factors** (one per RGB channel)
- Applied by simple **multiplication between pixel values and corresponding scalars**
- k instances of each filter type can be predicted per image

Graduated Filter



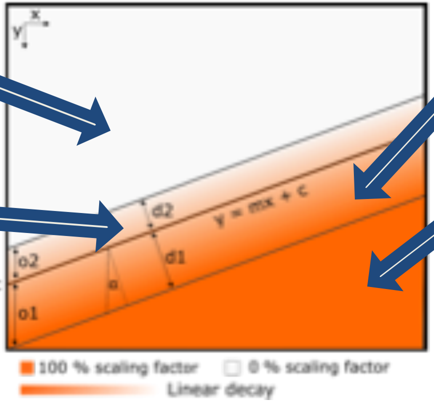


Elliptical Filter

Filter parametrisation

0% area : pixels are not adjusted

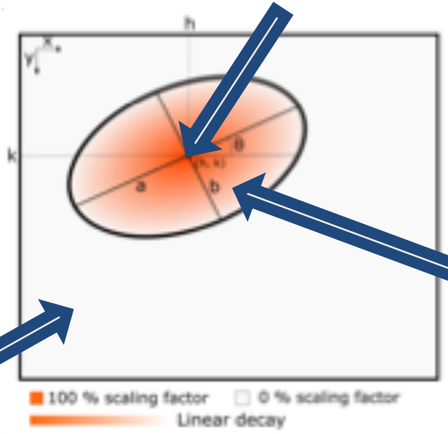
50-0% area : the scaling factor is further decreased linearly until 0%



100-50% area : the applied scaling factor linearly decreases from s_g to s_g / 2

100% area : adjust with scaling factor s_g

100% point : adjust with scaling factor s_e



Decay area : the scaling factor decreases linearly from s_e until 0

0% area : no changes

- Parameterized by three parallel lines
- The central line defines the filter location and orientation (+ Offsets o_1 and o_2)
- Binary parameter g_{inv} controls the filter inversion with respect to top and bottom lines.
- s^R_g , s^G_g , s^B_g are the 3 learnt scaling factors

- Defined by an ellipse
- Parameters : center (h, k), semi-major axis (a), semi-minor axis (b) and rotation angle (θ)
- s^R_e , s^G_e , s^B_e are the 3 learnt scaling factors

Filter parametrisation

Polynomial filter

- One Polynomial filter estimated per image
- **Emulates a brush tool**: flexible shapes with intrinsic **spatial smoothness**
- **Not instantiated as a heatmap of scaling factors** but as a **polynomial function** mapping the pixel intensity to a new value
- We consider order-p polynomial filters of the forms $\mathbf{i \cdot (x + y + \gamma)^{p}}$ and $\mathbf{(x + y + i + \gamma)^{p}}$, where i is the image channel intensity at pixel location (x, y), and γ is an independent scalar

Cubic-10 and Cubic-20

Polynomial filter

- After experiments, we find a **cubic polynomial** (p = 3) to offer both **expressive** image adjustments yet only a **limited set of parameters**
- We explore **two variants** of the cubic filter, c**ubic-10 and cubic-20**

$$
\begin{aligned}
i'(x,y) &= f(x,y,i) \\
&= i * (Ax^3 + Bx^2y + Cx^2 + D+ \\
&+ Dxy^2 + Exy + Fx + Gy^3 + Hy^2 \\
&+ Iy + J)
\end{aligned}
$$

Cubic-10

$$
\begin{aligned}
i'(x,y) &= f(x,y,i) \\
&= Ax^3 + Bx^2y + Cx^2i + Dx^2 + Exy^2 \\
&+ Fxyi + Gxy + Hxi^2 + Ixi + Jx \\
&+ Ky^3 + Ly^2i + My^2 + Nyi^2 + Oyi \\
&+ Py + Qi^3 + Ri^2 + Si + T
\end{aligned}
$$

Cubic-20

Parameter summary

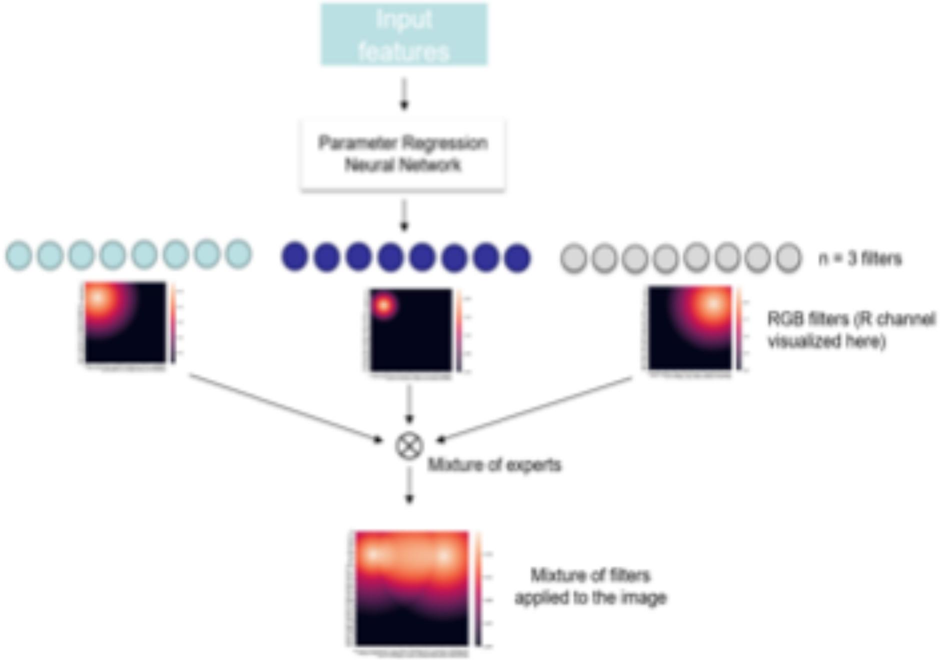
Parameters describing the three filter types

Filter	# Parameters	Parameters
Graduated s R	G=8	s^R_g , s^G_g , s^B_g, m, c, o_1, o_2, g_{inv}
Elliptical	E=8	s^R_e , s^G_e , s^B_e, h, k, θ, a, b
Cubic-10	P=30	{A · · · J} per colour channel
Cubic-20	P=60	{A · · · T} per colour channel

Fusing multiple filter instances

- Multiple instances of the **same filter type** : **Element-wise multiplication** resulting in a final heatmap

- Fusion of **graduated and elliptical** heatmaps : Simple **addition**



Loss Function

- Chrominance and luminance information split into two separate loss terms
- Focus on both local (MS-SSIM) and global enhancement (L1) operations during training

Given N image pairs $\{(Y_i, \hat{Y}_i)\}_{i=1\ldots N}$, where Y_i is the reference image and $\hat{Y}_i$ is the predicted image, we define the DeepLPF training loss function as:

$$\mathcal{L} = \sum_{i=1}^{N} \{\omega_{\text{lab}} \underbrace{||Lab(\hat{Y}_i) - Lab(Y_i)||_1}_{\text{Local enhancement}} + \omega_{\text{ms-ssim}} \underbrace{\text{MS-SSIM}(L(\hat{Y}_i), L(Y_i))}_{\text{Global enhancement}}\}$$

- Lab(·) : CIELab Lab channels of the input image
- L(·) : L channel of the image in CIELab colour space
- MS-SSIM : Multi-Scale Structural Similarity [26]

Experiments

Datasets

- **MIT-Adobe-5K dataset** [2]:
  - 5000 images captured using various DSLR cameras.
  - Each captured image is subsequently (independently) retouched by five human artists.
  - The image retouching of Artist C is used to define image enhancement ground truth.
  - **MIT-Adobe-5K-DPE** [3]: Subset used by DeepPhotoEnhancer (DPE) [3] with their dataset pre-processing procedure.
  - **MIT-Adobe-5K-UPE** [4]: Pre-processing following the protocol of DeepUPE [4].

- **See-in-the-dark (SID)** [5]:
  - 5094 image pairs, captured by a Fuji camera
  - Inputs are short-exposure images in raw format and the ground truth targets are long-exposure RGB images.
  - Both indoor and outdoor environments

Experiments

Evaluation metrics

- **PSNR** & **SSIM** : Standard metrics measuring the quality of the performed enhancement compared to the ground-truth image
- **LPIPS** [10] : Computes the distance between the features extracted from the enhanced image and ground truth using a backbone neural network. High-level metric correlated to human perception
- **Number of neural weights involved** : Accounts for the model efficiency

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2$$

$$PSNR = 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$

$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

$$\mathrm{SSIM}(x,y) = \left[l(x,y)^\alpha \cdot c(x,y)^\beta \cdot s(x,y)^\gamma\right]$$

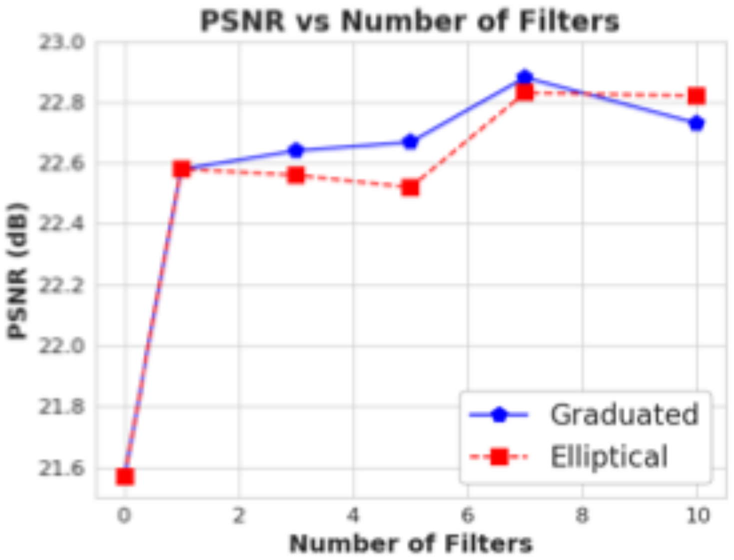
Quantitative results

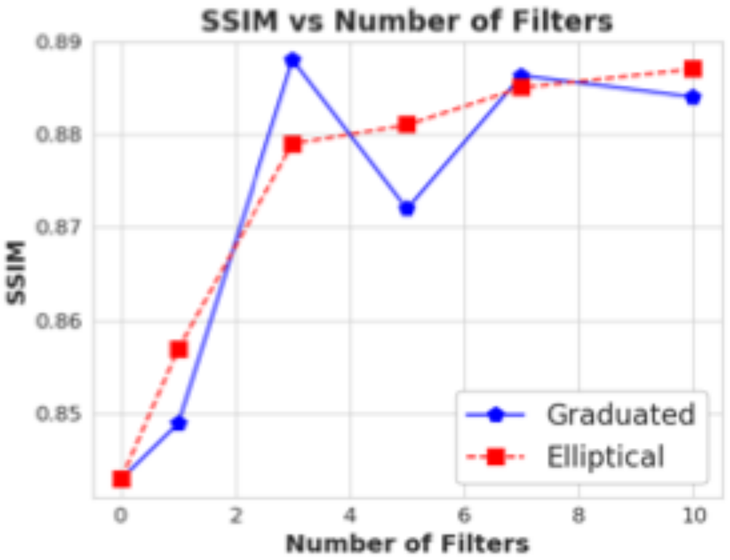
Ablation Study on the MIT Adobe-5K-DPE dataset

Architecture	PSNR↑	SSIM↑	LPIPS↓	# Weights
U-Net	21.57	0.843	0.601	1.3 M
U-Net+Elliptical	22.56	0.879	–	1.5 M
U-Net+Graduated	22.64	0.888	–	1.5 M
U-Net+Elliptical+Graduated	22.88	0.886	–	1.6 M
U-Net+Cubic-10	22.69	0.871	–	1.5 M
U-Net+Cubic-20	23.44	0.886	–	1.5 M
U-Net+Cubic-20+Elliptical+Graduated	**23.93**	**0.903**	**0.582**	1.8 M

Quantitative results

Ablation Study on the MIT Adobe-5K-DPE dataset



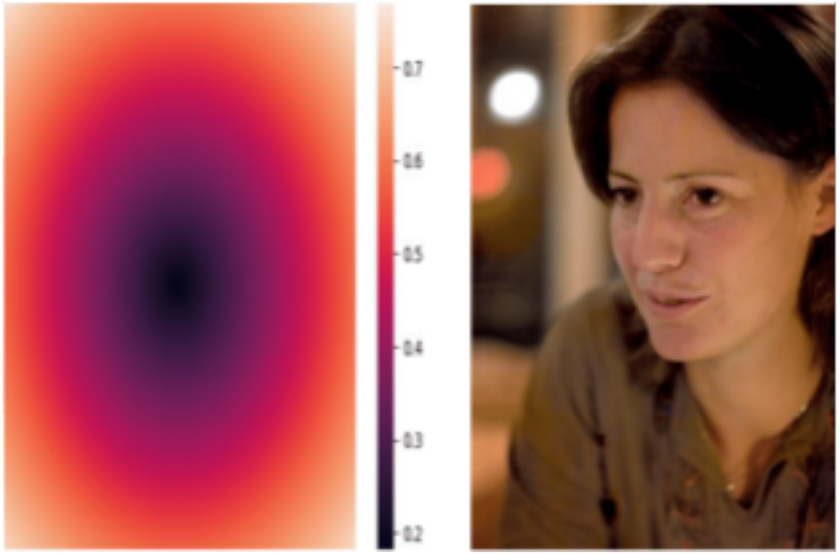


Qualitative results: predicted filters

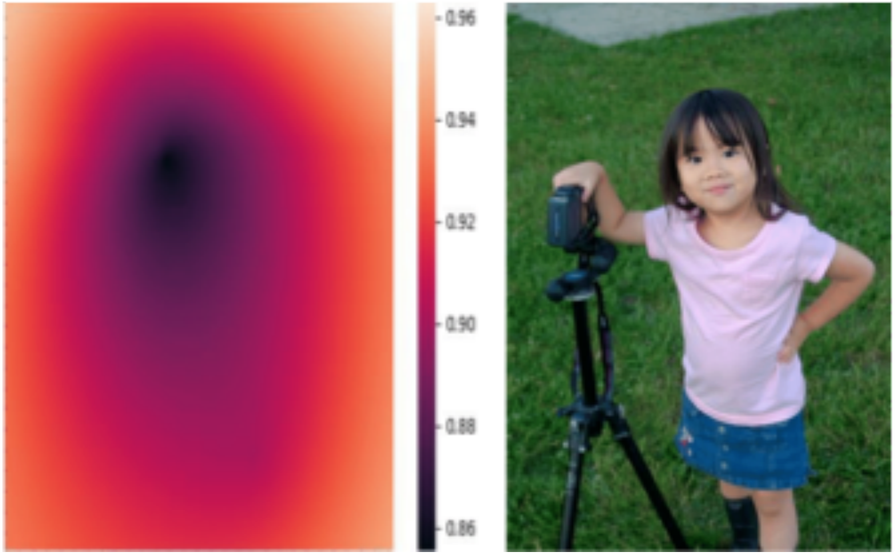
Graduated filter

Qualitative results: predicted filters

Elliptical filter



Mixture of Elliptical filters



Qualitative results: predicted filters

Cubic (Polynomial) filter

Quantitative results

Comparison to State of the Art

Architecture	PSNR↑	SSIM↑	LPIPS↓	# Weights
MIT-Adobe-5K-DPE dataset [3]				
DeepLPF	**23.93**	**0.903**	**0.582**	**1.8 M**
DPE [3]	23.80	0.900	0.587	3.3 M
MIT-Adobe-5K-UPE dataset [4]				
DeepLPF	**24.48**	0.887	**0.103**	**800K**
DPE [3]	22.15	0.850	–	3.3 M
DeepUPE [4]	23.04	**0.893**	0.158	1.0 M
SID dataset [5]				
DeepLPF	**26.82**	**0.702**	**0.564**	**2.0 M**
U-Net [5]	26.61	0.680	0.586	7.8 M

Qualitative Results - MIT Adobe-5K-DPE dataset

Input

DPE [3]

DeepLPF

Ground Truth

Qualitative Results - MIT Adobe-5K-DPE dataset

Input

DPE [3]

Ground Truth

DeepLPF

CLHE [6]

DPED (iphone7) [7]

NPEA [8]

FLLF [9]

Qualitative Results - MIT Adobe-5K-UPE dataset

Input

DeepUPE [4]

DeepLPF

Ground Truth

Qualitative results – SID dataset

Input

SID (U-Net) [5]

DeepLPF

Ground Truth

Summary

- DeepLPF : A **novel neural architecture** regressing the parameters of new **learnable filters** to be applied to an input image in order to retouch it
- Our filters emulate famous professional retouching tools and belong to 3 categories :
  - **Elliptical** filters : equivalent to the Radial filters in e.g. Lightroom or Photoshop
  - **Graduated** filters : emulating the professional tool with the same name
  - **Polynomial** filter : a regularized and locally smooth version of the brush tool
- Learnable parameterized filters allow
  - **intrinsic regularization** due to their expression
  - **interpretable** adjustments
- Our **filter fusion mechanism** allows to combine several instances of the same filter type and filters belonging to diverse categories in a given image
- We achieve state-of-the-art performance on 3 challenging benchmarks with a few amount of neural weights

References

[1] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention (MICCAI), volume 9351 of LNCS, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 6, 8

[2] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Fredo ´ Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In CVPR, pages 97– 104. IEEE, 2011. 1, 2

[3] Yu-Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and YungYu Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with GANs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6306–6314, 2018. 2, 4, 6, 7, 8

[4] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. Underexposed photo enhancement using deep illumination estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6849–6857, 2019. 2, 6, 7, 8

[5] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3291–3300, 2018. 2, 6, 7, 8

References

[6] S. Wang, W. Cho, J. Jang, M. Abidi, and J. Paik. Contrast-dependent saturation adjustment for outdoor image enhancement. Journal of the Optical Society of America A, 2017. 12, 17, 18

[7] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, pages 3277–3285, 2017. 2, 3, 7

[8] Y. Gao, H. Hu, and B. Li, and Q. Guo. Naturalness Preserved Non-Uniform Illumination Estimation for Image Enhancement Based on Retinex. IEEE Transactions on Multimedia, 2017. 12, 17, 18

[9] M. Aubry, S. Paris, S. Hasinoff, J. Kautz, and F. Durand. Fast Local Laplacian Filters: Theory and Applications. ACM Trans. Graph., 2014. 2, 12, 17, 18

[10] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In CVPR, 2018. 6