

Generative Quanta Color Imaging

Supplementary Material

A1 - Proof of Theorem 3.1

Given a 1bQIS image $\mathbf{Y}(u)$ where $u \in \mathcal{U}$ and $\mathcal{U} \subset \mathbb{Z}^2$, we have the local bit density $\mu \propto \|x\|_2$, where $\|\mathbf{Y}\|_2^2 := \sum_{u' \in N_u} \mathbf{Y}(u - u')^2$, and N_u is a local Euclidean grid centered at u . Then the output of ℓ -th layer of CNN layer can be written as,

$$\mathbf{Y}^\ell(u) = \sigma\left(\sum_{k=1}^m \phi_k^\ell \langle \mathbf{x}^{\ell-1}, \mathbf{\Lambda}^{\ell,k} \rangle_{N_u} + b^\ell\right), \quad (1)$$

where σ is nonlinear mapping such as ReLU which is non-expansive, $\mathbf{\Lambda}^{\ell,k}$ represents k -th atoms in ℓ -th layer, ϕ_k^ℓ corresponds to k -th coefficient, and b^ℓ is the bias of ℓ^{th} layer.

Theorem .1 *If $\mathbf{\Lambda}_1$ and $\mathbf{\Lambda}_2$ are filter atoms generated with neural ODE using integration intervals $(\tilde{\theta}_0, \tilde{\theta}_1)$ and $(\tilde{\theta}_0, \tilde{\theta}_2)$, respectively such that $\|\mathbf{\Lambda}_1 - \mathbf{\Lambda}_2\| \leq \epsilon |\tilde{\theta}_1 - \tilde{\theta}_2|$, for $\epsilon > 0$. If θ_1 and θ_2 are the exposures at $\tilde{\theta}_1$ and $\tilde{\theta}_2$, and activation function σ is non-expansive, then change in quanta exposure $\Delta\theta$ is continuous in filter atoms.*

Proof. As the result of atom-coefficient decomposition, a convolutional kernel of ℓ^{th} layer is $f_\ell = \phi^\ell \mathbf{\Lambda}^\ell(\tilde{\theta})$, where $\mathbf{\Lambda}^\ell(\tilde{\theta})$ corresponds to the atoms responsible for generating image with exposure $\tilde{\theta}$ and ϕ^ℓ corresponds to coefficients. The result of convolution operation on feature map $\mathbf{Y}^{\ell-1}$ from previous layer is given by -

$$\mathbf{Y}^\ell = f_\ell(\mathbf{Y}^{\ell-1}, \mathbf{\Lambda}^\ell(\tilde{\theta}), \phi^i) = \sigma\left(\sum_{k=1}^m \phi_k^\ell \langle \mathbf{x}^{\ell-1}, \mathbf{\Lambda}^{\ell,k} \rangle_{N_u} + b^\ell\right). \quad (2)$$

With sigma being non-expansive we can write (ignoring bias b_ℓ for simplicity) -

$$|\mathbf{Y}_1^\ell(u) - \mathbf{Y}_2^\ell(u)| \leq \left| \sum_{i=1}^k \phi_i^\ell \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) \rangle_{N_u} - \sum_{i=1}^k \phi_i^\ell \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2) \rangle_{N_u} \right| \leq \sum_{i=1}^k |\phi_i^\ell| \left| \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_1^{\ell-1}(\tilde{\theta}_1) \rangle_{N_u} - \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2) \rangle_{N_u} \right|, \quad (3)$$

let $B^\ell = [\langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) \rangle_{N_u} - \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2) \rangle_{N_u}]$. Then we have,

$$|\mathbf{Y}_1^\ell(u) - \mathbf{Y}_2^\ell(u)| \leq \sum_{i=1}^k |\mathbf{C}_i^\ell B_i^\ell|, \quad (4)$$

Using Hölder's inequality with $p = 2$ and $q = 2$ -

$$\begin{aligned} |\mathbf{Y}_1^\ell(u) - \mathbf{Y}_2^\ell(u)| &\leq \left(\sum_{i=1}^k (|\phi_i^\ell|^2)^{\frac{1}{2}} \right) \sum_{k=1}^k (|B_i^\ell|^2)^{\frac{1}{2}}, \\ &\leq \|\phi^\ell\|_2 \sum_{i=1}^k (|B_i^\ell|^2)^{\frac{1}{2}}. \end{aligned} \quad (5)$$

Consider only 2nd term in above equation $\sum_{i=1}^k (|B_i^\ell|^2)^{\frac{1}{2}} = \sum_{i=1}^k (|\langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) \rangle_{N_u} - \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2) \rangle_{N_u}|^2)^{\frac{1}{2}}$. Then,

$$|\langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) \rangle_{N_u} - \langle \mathbf{Y}^{\ell-1}, \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2) \rangle_{N_u}|^2 \leq |\langle \mathbf{Y}^{\ell-1}, (\mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) - \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2)) \rangle_{N_u}|^2. \quad (6)$$

Using Cauchy-Schwarz inequality -

$$|\langle \mathbf{Y}^{\ell-1}, (\mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) - \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2)) \rangle_{N_u}| \leq \|\mathbf{Y}^{\ell-1}\|_{2, N_u} \cdot \|\mathbf{\Lambda}_1^\ell(\tilde{\theta}_1) - \mathbf{\Lambda}_2^\ell(\tilde{\theta}_2)\|_2. \quad (7)$$

Substituting results from Eq. 5 and 6 into Eq. 4 -

$$\begin{aligned}
|\mathbf{Y}_1^\ell(u) - \mathbf{Y}_2^\ell(u)| &\leq \|\phi^\ell\|_2 \cdot \sum_{i=1}^k \left(\|\mathbf{Y}^{\ell-1}\|_{2, N_u}^2 \cdot \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2^2 \right)^{\frac{1}{2}}, \\
&\leq \|\phi^\ell\|_2 \|\mathbf{Y}^{\ell-1}\|_{2, N_u} \sum_{i=1}^k \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2.
\end{aligned} \tag{8}$$

$$\sum_{u \in \mathcal{S}} |\mathbf{Y}_1^\ell(u) - \mathbf{Y}_2^\ell(u)|^2 \leq \|\phi^\ell\|_2^2 \|\mathbf{Y}^{\ell-1}\|_{2, N_u}^2 \sum_u \sum_{i=1}^k \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2^2. \tag{9}$$

Note that -

$$\sum_u \sum_{i=1}^k \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2 = \sqrt{|\mathcal{U}|} \cdot \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2, \tag{10}$$

with $|\mathcal{U}|$ being the domain of $\mathbf{Y}^{\ell-1}$. Then Eq.8 becomes -

$$\|\mathbf{Y}_1^\ell - \mathbf{Y}_2^\ell\|_2 \leq \|\phi^\ell\|_2 \|\mathbf{Y}^{\ell-1}\|_{2, N_u} \sqrt{|\mathcal{U}|} \cdot \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2. \tag{11}$$

Since the bit density of 1bQIS is the average number of ones in the space-time volume. Hence $\|\mathbf{Y}\|_2 \propto \mu$. Then,

$$\|\mu_1^\ell - \mu_2^\ell\|_2 \leq \|\phi^\ell\|_2 \|\mathbf{Y}^{\ell-1}\|_{2, N_u} \sqrt{|\mathcal{U}|} \cdot \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2. \tag{12}$$

Since bit density is the sufficient statistic of exposure θ , We can write -

$$\|\theta_1^\ell - \theta_2^\ell\|_2 \leq \|\phi^\ell\|_2 \|\mathbf{Y}^{\ell-1}\|_{2, N_u} \sqrt{|\mathcal{U}|} \cdot \|\Lambda_1^\ell(\tilde{\theta}_1) - \Lambda_2^\ell(\tilde{\theta}_2)\|_2. \tag{13}$$

A2 - Architecture, Hyper-parameters, Synthetic Data Generation and Benchmarks

A2.0 - Details About Comparison With Image Augmentation Approach

For the results presented in ??, we compare the performance of image augmentation without exposure correction against our method. During the training phase of the colorizer, we augment the dataset with images from an exposure spectrum of $\tau = \{5, 10, 15\}$. However, for the exposure correction-based approach, we use the same exposure levels but do not perform dataset augmentation. Both methods are trained for similar number of iterations and tested on images with an exposure of $\tau = 0$ to ensure fair comparison.

A2.1 - Architecture

In this section, we detail the architectures used for exposure correction, Single Image Colorization (SIC) and Burst Image Colorization (BIC).

A2.1.1 - AtomODE Pix2Pix and AtomODE Cycle

The AtomODE Pix2Pix and AtomODE Cycle are modified versions of Pix2Pix [1] and CycleGAN [4] respectively. The key difference between Pix2Pix and AtomODE Pix2Pix lies in the bottleneck layer of the generator. We substitute the convolutional layers in the generator with Exposure Adaptive Convolutional Layer (EACL). For NeuralODE we design 6 layers of [GroupNorm, Activation, Convolutional Layer]. We use Tanh as activation function. We use $\text{rtol} = 1 \times 10^{-3}$ and $\text{atol} = 1 \times 10^{-3}$. We use a multiscale discriminator for both AtomODE Pix2Pix and AtomODE Cycle. Further, we provide a detailed account of the hyperparameters in Section S3.2.

A2.1.2 - Pix2PixFuse

Pix2PixFuse inherits the meta-architecture from Pix2Pix with an additional Cross Non-Local Fusion (CNLF) [2] block. We use CNLF block to constructively fuse the representations obtained from images of different exposures. In Figure 1 shows the Pix2PixFuse architecture.

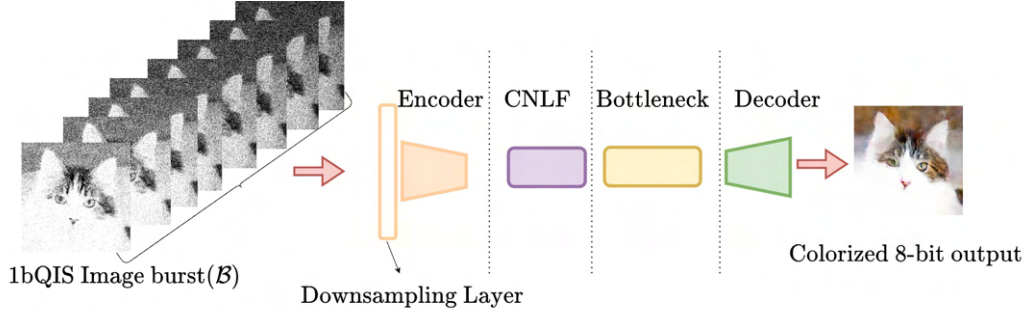


Figure 1. Pix2PixFuse for burst image colorization.

Hyperparameters	Exposure Correction Methods	
	AtomODE Pix2Pix	AtomODE CycleGAN
Input image dimension (# of images, #channels, height, width)	1, 1, 512, 512 or 1,1,256, 256	1, 1, 512, 512 or 1,1,256,256
learning rate	0.00002	0.00002
norm	BatchNorm and GroupNorm(Neural ODE)	batch and GroupNorm(Neural ODE)
dropout rate	0.5	0.5
batchsize	1	1
input number of channels	1	1
output number of channels	1	1
serial batches	True	False
no-flip	True	True
netG	ResNet-9	ResNet-9
# of filter in first layer of generator	64	64
# of blocks in generator	9	9
number of epochs	100	100
optimizer	Adam	Adam
betas	(0.5, 0.999)	(0.5, 0.999)
Discriminator type	Multiscale	Multiscale
# of discriminators	2	2
# of layers in discriminators	3	3
# of filter in first layer of discriminator	64	64
weight for feature matching loss	0	0
Adversarial loss	True	True
VGG loss	False	False
GAN type	Vanilla GAN	Vanilla GAN
activation	sigmoid and Tanh (only for Neural ODE)	sigmoid and Tanh (only for Neural ODE)

Table 1. Hyperparameter table for AtomODE Pix2Pix, AtomODE CycleGAN.

A2.2 - Hyper-parameters

The hyperparameters used by different architectures are shown in Table 1 and Table 2. AtomODE Pix2Pix and AtomODE CycleGAN are trained on 24GB Nvidia 3090Ti GPU and rest of the models are trained on V100.

A2.3 - Additional Details About Benchmarks

In this section, we provide additional training details about the benchmarks we used to compare our algorithm.

- **DLOW**: DLOW is a continuous model by design. We train it with $\tau = 0$ exposure image as the input domain and $\tau = 14$ as the target domain. We supply images from the intermediate domains $\tau = 4, 9$ for additional guidance. The model was trained for 100 epochs with a learning rate of 0.0002, batch size 1. DLOW uses cycleGAN architecture and is trained using unpaired data setting.
- **Pix2PixDNI**: Deep Network Interpolation to interpolate among kernels of finetuned networks for continuous i2i. We train two models that translate input image $\tau = 0$ to $\tau = 9$ and from $\tau = 0$ to $\tau = 14$. We interpolate between these two models to obtain multiple models that translate input ($\tau = 0$) to $\tau = \{k\}_{k=1}^4$. We use Pix2Pix architecture using paired data setting. All the other hyper-paramters remain similar to AtomODE Pix2Pix.
- **CycleDNI**: Deep Network Interpolation to interpolate among kernels of finetuned networks for continuous i2i. We train two models that translate input image $\tau = 0$ to $\tau = 9$ and from $\tau = 0$ to $\tau = 14$. We interpolate between these two models

Hyperparameters	Single Image Colorizer	Burst Image Colorzier
	Pix2Pix	Pix2PixFuse
Input image dimension (# of images, #channels, height, width)	1, 1, 512, 512 or 1,1,256,256	N, 1, 512, 512
learning rate	0.0002	0.0002
norm	BatchNorm	BatchNorm
dropout rate	0.5	0.5
batchsize	6	1
input number of channels	1	1
output number of channels	3	3
serial batches	True	True
no-flip	True	True
netG	ResNet-9	ResNet-9 + CNLF
# of filter in first layer of generator	64	64
# of blocks in generator	9	9
number of epochs	250	100
optimizer	Adam	Adam
betas	(0.5, 0.999)	(0.5, 0.999)
Discriminator type	Multiscale	Multiscale
# of discriminators	2	2
# of layers in discriminators	3	3
# of filter in first layer of discriminator	64	64
weight for feature matching loss	0	0
Adverserial loss	True	True
VGG loss	True	True
GAN type	Vanilla GAN	Vanilla GAN
activation	sigmoid	sigmoid

Table 2. Hyperparameter table for SIC and BIC architectures.

to obtain multiple models that translate input ($\tau = 0$) to $\tau = \{k\}_{k=1}^4$. We use CycleGAN architecture using unpaired data setting. All the other hyper-paramters remain similar to AtomODE CycleGAN.

- **SAVI2I** : SAVI2I synthesizes intermediate results between two domains. In our case, the input domain image belongs to $\tau = 0$ and output domain $\tau = 14$. We do not use style reconstruction loss as it led to inferior results.

A2.4 - Synthetic Data Generation

In this section, we detail the generation of 1bQIS versions of AFHQ and CelebA datasets from their RGB counterpart. In Figure 2, we show a simplified imaging model of 1bQIS. Following are the steps followed for generating burst of 1bQIS images with varying exposures from a single RGB image.

- **Step 1** : Extract quanta exposure θ from RGB images.
- **Step 2** : We simulate exposure bracketing to generate burst of 1bQIS images with varying burst. To this end, we divide the overall exposure (θ) into brackets. The bracketing is controlled using the parameter α . α varies between (1, 25) with a difference of 0.5, set of all such $\alpha = \{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0\}$. Let the exposure obtained by dividing α_i be θ_i . In summary we obtain a set of exposures $\{\theta_i\}_{i=0}^{14}$ using set of $\{\alpha_i\}_{i=0}^{14}$.
- **Step 3** : Using the set of α in Step 2, obtain the binary measurements following the imaging model of 1bQIS. If $\mathbf{X} = \text{Poisson}(\theta_i) + \text{Gaussian}(0, \sigma^2)$ be the measurements before binarization and let \mathbf{Y} denote the binarized output. Y is assigned a value of “1” if $\mathbf{X} \geq q$ else “0”. We use $q = 0.5$ to generate the synthetic data.
- **Step 4** : In the synthetic dataset, we have an exposure burst going from overexposed \rightarrow underexposed for each image. There is a total of 15 images in each burst and we show one such burst in Figure 3. All the images in a burst are indexed using an indexing variable τ . We assign a continuous variable $\tilde{\theta} \in (0, 1)$. For example, the assigned continuous variable $\tilde{\theta}$ for the example exposure spectrum is shown in Figure 3 is - $\{0.0625, 0.125, 0.1875, 0.25, 0.3125, 0.375, 0.4375, 0.5, 0.5625, 0.625, 0.6875, 0.75, 0.8125, 0.875, 0.9375\}$

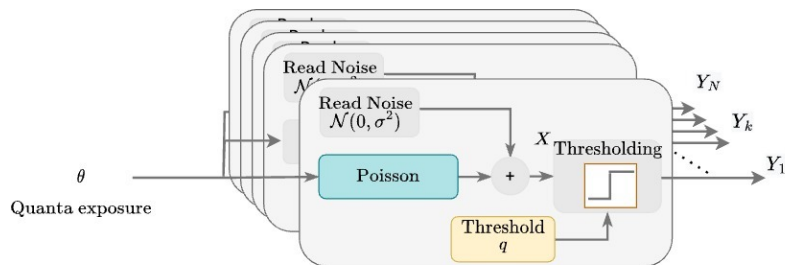


Figure 2. 1bQIS imaging model showing multiple output binary images of the same scene.

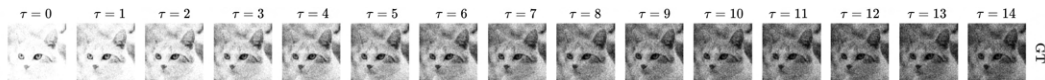


Figure 3. An example exposure burst.

A3 - Assumptions and Drawbacks

We make following assumptions for our method to work -

- In the main paper, we showed the burst recovery using an overexposed input image. This model expects an overexposed ($0.9 \geq \text{bit density} \geq 0.75$) image as input to recover the burst with high accuracy.
- We assume that representative images of the burst are available for training as neural ODEs cannot extrapolate accurately.
- We assume that the continuous label $\tilde{\theta}$ of input image and target image are known beforehand.

The drawbacks of our method are as follows -

- Generative Adversarial Networks are known to be hard to optimize and converge. Since we adopt GAN framework, we inherit the optimization and convergence issues. For our experiments, we experimented with lower learning rates to stabilize the training process.
- In the main paper, we used Pix2Pix and CycleGAN as parent networks to derive our methods. These variations of GANs are not known for the best output image fidelity. Hence, we see several artefacts in the generated images. Using latest techniques and improvements in GAN literature we can achieve better fidelity especially for 512 x 512 experiments.

A4 - Additional Qualitative Results on Synthetic Data

In this section, we show additional qualitative results for exposure burst recovery and colorization. First, in Section S5.1 we show examples of exposure bursts recovered using our method for cases where input is (a) an over-exposed image (b) an under-exposed image.

A4.1 - Recovered Exposure Spectrum

- **Case 1 - With overexposed binary image input** - In Figure 4 we show the results for burst recovery with an overexposed binary image as input.
- **Case 2 - With underexposed binary image input** - In Figure 5 we show the results for burst recovery with an underexposed binary image as input and colorization results in Figure 6.

A4.2 - Additional SIC Results and Colorization results for Overexposed Input

In Figure 7, we show some additional qualitative results for SIC for our methods, AtomODE Pix2Pix, AtomODE CycleGAN and competing methods.

A5 - Experiment Setup for Real World Image Capture

This section describes the experimental setup we build to capture the real-world images used in this paper. As in Figure 8, our setup consists of a monochrome camera that is either a CMOS (Basler ace acA2440-35um) or a QIS (Gigajot PathFinder prototype camera [3]), an RGB camera (Basler ace acA2440-35uc), and a flat target that contains a pattern from the AFHQ dataset and a series of April Tags. We take images of the target using both cameras and warp the RGB image to align with the

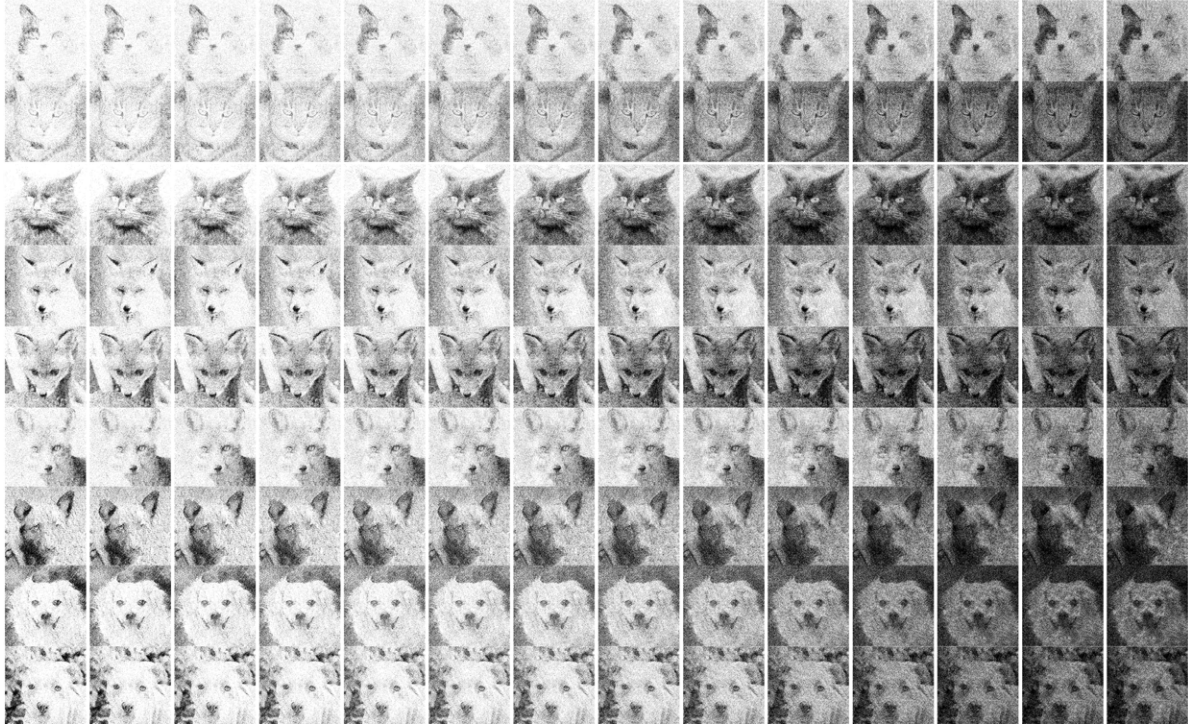


Figure 4. Examples of recovered burst from overexposed input for AtomODE CycleGAN.

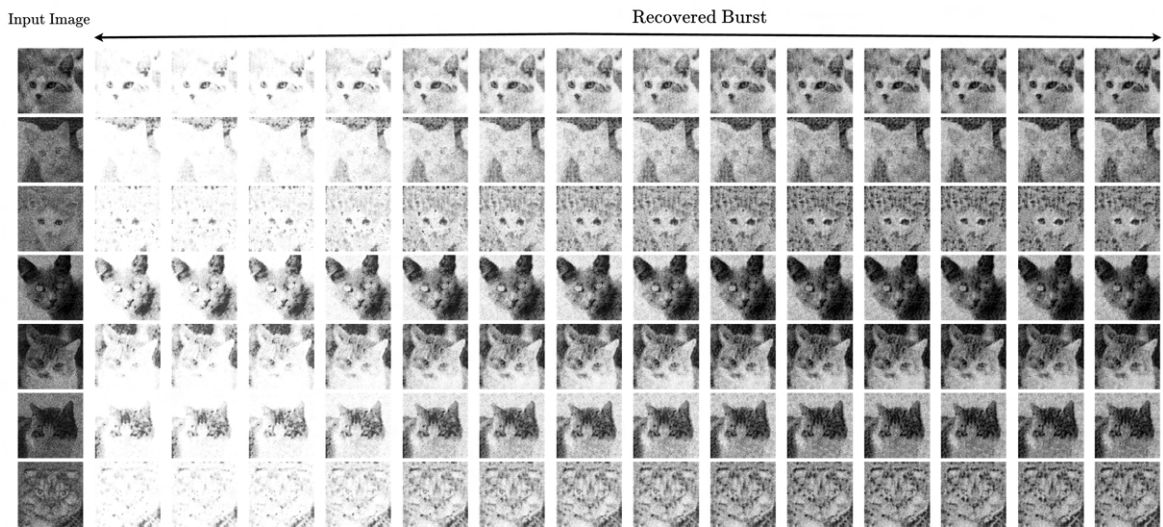


Figure 5. Examples of recovered burst from underexposed input for AtomODE Pix2Pix.

monochrome image by calculating the homography from the detected April Tags. When mounting the setup, we ensure that the monochrome camera is approximately front parallel to the target, so that the pattern is rectangular in the monochrome image and can be naively cropped out. This is because we want to avoid warping the monochrome image; the interpolation required by warping can alter the statistical property of the pixel values of the warped image.

We use the following parameters in our image capture process. The CMOS monochrome camera was set with an exposure time of 464 μ s, a gain of 24 dB, a digital shift of 0, and a pixel format of Mono 12. We noticed that when the gain is 24 dB, each gray level of the pixel value corresponds to one excited electron on the photodiode. The RGB camera was set with an



Figure 6. Examples of colorized image from underexposed input for using AtomODE Pix2Pix as exposure correction method and Pix2PixHD as colorizer.

exposure time of 30000 us, a gain of 0 dB, a digital shift of 0, and a pixel format of Bayer RG 12. Lastly, the QIS camera was set with an exposure time of 74 us, a gain of 0 dB, a digital shift of 0, and a pixel format of Mono 14.

To fine-tune and test our model on real data, we capture a total of 125 image pairs, among which 100 are for fine-tuning and 25 are for testing. For each fine-tuning and testing pair, we use different targets with patterns sampled from the training and testing set of AFHQ dataset, respectively. Each image pair i for fine-tuning consists of a CMOS monochrome image I_i^{Mono} and an RGB image I_i^{RGB} of the same target captured using the previously stated camera settings, $i = 1, \dots, 100$. Similarly, for the testing data, we collect 25 pairs of CMOS monochrome images I_i^{Mono} and RGB images I_i^{RGB} , $i = 101, \dots, 125$, as well as 25 pairs of QIS images I_i^{QIS} and RGB images I_i^{RGB} , $i = 126, \dots, 150$.

After the data capture, we need to translate the pixel value of each image to represent the number of photons received by each pixel. For the monochrome CMOS images I_i^{Mono} , we can relate the gray level of each pixel I to the number of photons using the following formula:

$$X = G \times I / QE. \quad (14)$$

Here, X is the number of photons recorded by a pixel in an exposure period plus the dark noise, G is the value of the gain ratio, I is the value of the pixel, and QE represents quantum efficiency. Based on our previous discussion, in our setup, the value of G is 1 in our experiment. The quantum efficiency QE is 0.68 according to the datasheet. By applying Equation 14, we can convert the pixel value of each CMOS image I_i^{Mono} to the number of photons received in each pixel X .

As for the QIS camera, we first calibrate the dark current I_{dark} and the photo response non-uniformity (PRNU) function CRF() on the raw data. Subsequently, we convert the calibrated data to photon counts by taking into account the quantum efficiency (QE) based on the following equation:

$$I = \text{ADC}\{\text{Clip}\{G \times \text{Poisson}(ET \times QE \times (\text{CRF}(X) + I_{\text{dark}}))\}\} + \text{Gaussian}(0, \sigma_{\text{real,noise}}^2), \quad (15)$$

where X is the number of photons received by a pixel in an exposure period, G is the gain ratio, I is the pixel value of the QIS camera, QE is quantum efficiency, ET is the exposure time, and ADC represents the analog to digital converter. We can directly obtain the number of photons X received by each pixel directly using the translation equation provided by the manufacturer [3].

A6 - Details about finetuning on real-world data

The results reported for real world CMOS images we use a finetuned colorizer. We first recover the exposure burst using AtomODE CycleGAN and use the image at index $\tau = 10$ as input 1bQIS image and corresponding CMOS RGB image as target image for finetuning. We finetune the colorizer for 50 epochs on 100 training image pairs (1bQIS and corresponding CMOS RGB image).



Figure 7. SIC results on AFHQ dataset using Pix2Pix colorizer.

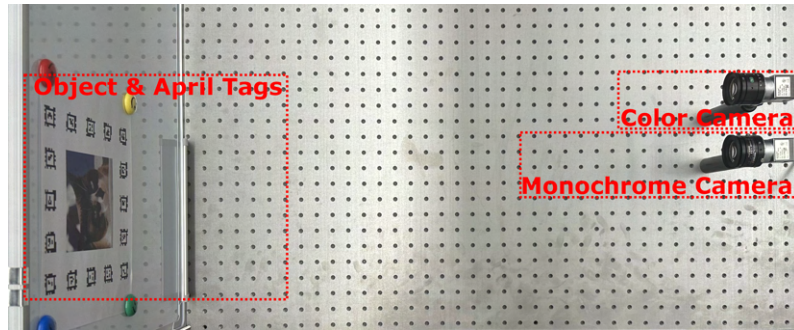


Figure 8. The experimental setup.

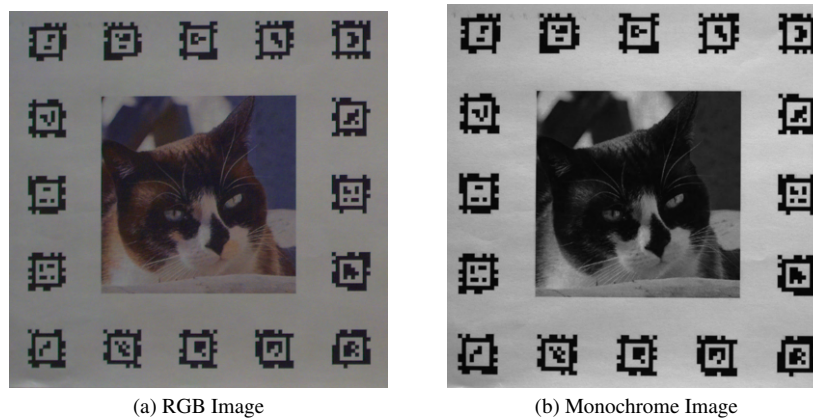


Figure 9. Sample images taken by the RGB camera and the monochrome camera without alignment.

References

- [1] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-Image translation with conditional adversarial networks. In *CVPR*, pages 5967–5976, 2017. [2](#)
- [2] Ziwei Luo, Lei Yu, Xuan Mo, Youwei Li, Lanpeng Jia, Haoqiang Fan, Jian Sun, and Shuaicheng Liu. EBSR: Feature Enhanced Burst Super-Resolution with deformable alignment. In *CVPR*, pages 471–478, 2021. [2](#)
- [3] Jiaju Ma, Saleh Masoodian, Dakota A Starkey, and Eric R. Fossum. Photon-number-resolving megapixel image sensor at room temperature without avalanche gain. *Optica*, 4(12):1474–1481, 2017. [5](#), [7](#)
- [4] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pages 2242–2251, 2017. [2](#)