arXiv:1909.10136v1 [eess.IV] 23 Sep 2019

LISR: Image Super-resolution under Hardware Constraints

Pravir Singh Gupta[1], Xin Yuan[2], Gwan Seong Choi[1]

Texas A&M University, College Station, Texas, USA[1]
Nokia-Bell Labs, Murray Hill, New Jersey, USA[2]
pravir@tamu.edu, xyuan@bell-labs.com, gwanchoi@tamu.edu

Abstract

We investigate the image super-resolution problem by considering the power savings and performance improvement in image acquisition devices. Toward this end, we develop a deep learning based reconstruction network for images compressed using *hardware-based downsampling, bit truncation and JPEG compression*, which to our best knowledge, is the first work proposed in the literature. This is motivated by the fact that binning and bit truncation can be performed on the commercially available image sensor itself and results in a huge reduction in raw data generated by the sensor. The combination of these steps will lead to high compression ratios and significant *power saving* with further advantages of image acquisition simplification. Bearing these concerns in mind, we propose LISR-net (Lossy Image Super-Resolution network) which provides better image restoration results than state-of-the-art super resolution networks under hardware constraints.

1 Introduction

We are living in a multimedia world. With the advent of Internet and mobile devices, the amount of multimedia content generated by users is increasing at a tremendous rate. In addition, with the improvement in VLSI (Very Large Scale Integration) technology, resolution of image sensors are also increasing. Smartphones with image sensor resolution greater than 20 Megapixel are commonly used and image sensor vendors are also offering sensors more than 40 Megapixel range. However, lately Moore's law has started to saturate and hence there is an increasing pressure to extract performance improvements from architectural and algorithmic innovations than device scaling. Videos and images present a huge burden in processing, storage and transmission networks. It also presents a challenge in terms of power consumption in image acquisition devices such as mobile devices.

On the other hand, the advanced machine learning (ML) techniques especially deep learning perform much better than traditional computer vision techniques for tasks like super-resolution, object detection *etc*. Inspired by this, this paper exploits the power of deep learning techniques to reduce the data traffic in the *image acquisition pipeline* which will make it easier to meet power and performance requirements. (He et al. 2016; Timofte et al. 2018). Motivated by the fact that if hardware constraints of imaging system are taken into consideration then deep neural networks (DNN)

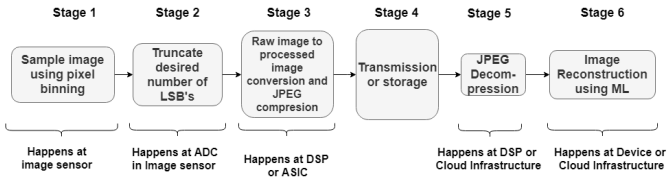


Figure 1: Proposed Image Acquisition Methodology: It consists of 6 stages. In the stages 1-3, an image gets compressed using downsampling, bit truncation and JPEG. Stage 4 represents transmission of image which can be a wireless medium or an on-chip bus or even storage. Stage 5 performs JPEG decompression and stage 6 consists of the proposed LISR-net to restore the desired image.

Figure 2: Left: 0703 image from the DIV2K dataset (Agustsson and Timofte 2017). Downsampling is performed via 2×2 binning, with bit truncation such that B.W. (bit width) = 7 and JPEG compression at quality $Q = 90$. Right: (i) HR patch from the left image, (ii) reconstruction using our proposed LISR-net, (iii) reconstruction using Bicubic interpolation, and (iv) reconstruction with EDSR (Lim et al. 2017).

will be able to address these problems of imaging systems, we propose image compression using combination of downsampling (binning), bit truncation and JPEG compression. To reconstruct the original image, we propose LISR-net, which is a DNN based reconstruction network to perform super-resolution (to restore loss of resolution caused by binning) and artifact removal caused by bit truncation and JPEG compression. The proposed sensing methodology is shown in Fig. 1, which is different from existing super-resolution networks, and one exemplar reconstruction is shown in Fig. 2.

While DNN has been used for super-resolution (Ref. (Dong et al. 2016; Kim, Kwon Lee, and Mu Lee 2016; Ledig et al. 2017; Wang, Chen, and Hoi 2019; Yang et al. 2018)), this work is novel in the following perspectives:

i) We propose a new image acquisition framework (Fig. 1), using bit truncation and JPEG compression along with downsampling, which is more realistic than existing methods, since from an imaging system perspective, lossy JPEG compression is the preferred way of image compression.

ii) Downsampling operation in our framework is *averaging and rounding* instead of bicubic which is more popular for superresolution tasks (Timofte et al. 2018). Again our method is more realistic as averaging operation is easy to implement in hardware especially at image sensor level using pixel binning technique.

Following this, our proposed sensing framework in Fig. 1 results in significant reduction in raw data rate *i.e*. data generated from image sensor as well as final image size. This has direct potential for power savings, transmission bandwidth savings and simplified hardware implementation. One can argue that instead of downsampling one can use a low resolution image sensor itself. This argument will hold valid only if reconstruction or super resolution process is exact, which is not possible. Thus downsampling using binning gives user an option to choose between the two depending upon requirements. There are some other works in the literature which have proposed pixel bit depth enhancements (Liu, Sun, and Liu 2017; Liu et al. 2019; Su et al. 2019). However these works were targeted at converting low bit depth images to high bit depth display. Another work proposed super-resolution and bitdepth enhancement (Umeda et al. 2019), however, the authors used A+ (Adjusted anchored neighborhood regression algorithm) (Timofte, De Smet, and Van Gool 2014) method for super-resolution instead of Neural Networks. Some works have also focused on denoising and compression artifact removal tasks (Mao, Shen, and Yang 2016; Yu et al. 2016; Zhang et al. 2017). To the best of our knowledge, this is the first work to investigate the DNN based image restoration considering the combination of downsampling, bit truncation and JPEG for compression and power savings.

2 Background

In this section, we review the background of building blocks of the proposed image acquisition methodology in Fig. 1. Via introducing the image sensors in Sec. 2.1, we understand why *binning* is preferred in hardware. By analyzing the power consumption in Sec. 2.2, we understand how to save power of the device using super resolution. The JPEG compression is reviewed in Sec. 2.3, and following this, the transmission and storage techniques are described in Sec. 2.4. As a widely used component in DNN, the residual network, which will be used in our design is introduced in Sec. 2.5.

2.1 Image sensors

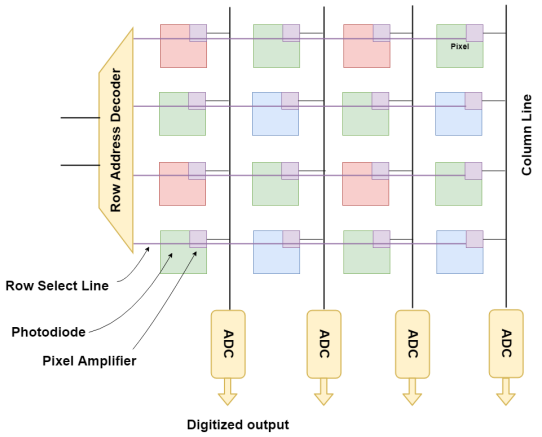


Figure 3: A simple schematic of image sensor.

An image sensor converts light intensity into electrical signals. It consists of a rectangular grid of pixels. Typically pixels are addressed in a row wise fashion *i.e*. all pixels in the same row are addressed simultaneously. The addressed row outputs the signal in column lines which is connected to Analog-to-Digital Converter (ADC). The ADC produces digitized version of the image. The number of digitized bits depend upon the resolution of image sensor. A simple schematic of image sensor is shown in Fig. 3.

Most commercially available image sensors have a feature called "binning" which simply means to *combine information of a group of neighboring pixels together to form a single pixel in the output image*. Binning results in loss of spatial resolution by a factor equal to the number of pixels binned together, which can be either an addition operation or an averaging operation on image pixels. Addition operation has the advantage in low light situations as it increases the low light sensitivity of image sensor (Huang, Conge, and Huang 2011); on the other hand, averaging operation has advantage in common lighting conditions as it prevents saturation of pixels in that scenario (Zhou, Pain, and Fossum 1997). Binning has been used in various applications successfully like low light imaging (Huang, Conge, and Huang 2011), background noise suppression (Cho et al. 2013), power reduction (Kumagai et al. 2018), and multiresolution (Yoshihara et al. 2006; Zhou, Pain, and Fossum 1997), *etc*. Since binning reduces the number of samples, it results in significant reduction in power consumption in an image sensor as ADC is responsible for major chunk of

power consumption (Oike and El Gamal 2013). By decreasing the *bit resolution* of ADC one can reduce the consumption exponentially. This is because noise and linearity requirements are relaxed at smaller bit resolutions (Yip and Chandrakasan 2013). In this paper, we use 2×1, 2×2 and 4×4 binning modes with averaging.

2.2 Power and performance analysis of digital circuits

Power consumption is one of the major concerns in digital circuits especially in mobile devices. The are two main independent components of power consumption/dissipation in a digital circuit - *Static power and Dynamic power*. Hence total power consumption can be written as

$$P_{total} = P_{static} + P_{dynamic}. \tag{1}$$

Static power is the power consumed when there is no activity in a digital circuit and *Dynamic power* is the power consumed due to switching signals in a digital circuit. Dynamic power can be further broken down as

$$P_{dynamic} = P_{switching} + P_{short\ circuit}, \tag{2}$$

where $P_{switching}$ refers to the power required to charge or discharge the switching nodes in digital circuit and $P_{short\ circuit}$ refers to transient power consumption due to short circuit current when a gate switches from one state to another.

$$P_{switching} = \alpha C V^2 F, \tag{3}$$

where α is switching activity factor *i.e.* , number of times a signal switches per cycle, F denotes the frequency, V represents the voltage and C denotes switched capacitance at the node. $P_{switching}$ is one of the major concerns when comes to power consumption in a digital circuit. One can see from Eq. (3) that power consumption is linearly proportional to frequency F and quadratically proportional to voltage V. In general frequency of digital system is determined by the data processing requirements *i.e.* , how much data must be processed per second. Thus if one wants to process half the amount of data in the same time, one can halve the operating frequency. Voltage and Frequency in Eq. (3) are not independent quantities but follow a proportional relationship. If the operating frequency is increased, the operating voltage must be increased and vice-versa due to device physics and noise margin requirements. Therefore, if there is a reduction in the data to be processed one can decrease voltage and frequency to achieve quadratic reduction in energy consumption. When this reduction in voltage and frequency is performed on the fly depending upon the data processing requirement it is called DVFS (Dynamic Voltage Frequency Scaling).

Another important factor when designing a digital system is *bitwidth* of the data. If the datapath is serial then it will imply longer time to transfer data and if it is parallel it will imply a wider data bus. Longer bitwidth implies larger arithmetic circuits such as adders, multipliers *etc*. or multiple clock cycles of operation. For instance, a single 4 bit adder can add two 8 bit numbers in 2 cycles or two 4 bit adder can add 8 bit numbers in one cycle. This increases the delay of the most critical path or the slowest path in the circuit resulting in slower operation. To speed up one might have to use faster and power hungry circuits.

While there are many other factors impacting power and performance, a detailed analysis of those are beyond the scope of this work and only relevant issues are discussed here.

2.3 JPEG

After the sensor captures the image, the image is compressed by some method. JPEG (Joint Photographics Expert Group) is one of the most widely used lossy compression technique for images. Though JPEG can perform both lossy and lossless compression, the former one is popular due to little loss of perceptual quality. This is because most of the image information is contained in a very few coefficients in the discrete cosine transform (DCT) domain and hence insignificant coefficients can be discarded without much loss in perceptual quality producing large compression ratios. One can also control the amount of loss (and hence compression) by using the 'Quality' parameter of JPEG. It ranges from $1 - 100$ with 100 being the best lossy compression one can achieve. JPEG generally performs DCT on a block of 8×8 pixels, followed by quantization of DCT coefficients which constitutes the lossy step. After quantization codec is performed such as Huffman coding, run length coding *etc*. DCT is generally the most energy consuming part in JPEG compression (Shichao, Zhizhong, and Xin 2016). Thus reduction in image data will lead to reduction in energy consumed in JPEG compression by a similar factor.

2.4 Consideration in transmission and storage

After the image is compressed, *e.g.* , via JPEG, into bit streams, in wireless transmission, the data is encoded using ECC (Error Correction Coding) schemes to tolerate the errors occurring in wireless transmission due to channel noise. ECC schemes provide some type of redundancy in data to detect and correct the errors. One of the most popular ECC is LDPC (Low Density Parity Check Codes). It is used in both storage (*e.g*. Solid State Drives (Luo et al. 2018)) and wireless transmission (*e.g*. 5G specifications (3GPP 2017)). LDPC uses parity check bits to detect and correct errors. A simple schematic of transmitted data packet is shown in Fig. 4. It consists of message packet plus ECC bits (Fig. 4(a)). If the channel is noisy then the transmitter would require to encode the message more strongly (Fig. 4(b)). Thus for a fixed transmitted data packet size, actual message packet would become smaller resulting in reduced message bandwidth. If there is a possibility of compressing the message data then, one can either send more message bits per data packet resulting in increase of message transmission bandwidth or one can encode message bits more aggressively to make it more resistant to channel noise while keeping the message transmission bandwidth the same (Fig. 4(b) and (c)). For storage the analysis it is analogous to that of wireless channel.

2.5 Deep residual networks

With the recent advances of ML, deep learning based algorithms have demonstrated superior performance than con-

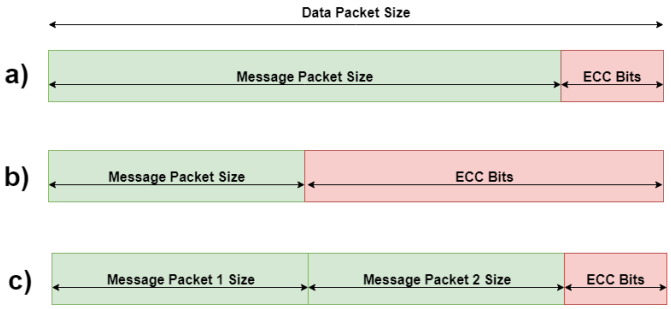


Figure 4: Packet size analysis in wireless transmission.

ventional methods for image super-resolution. Most of them are based on convolution neural networks (CNNs), which apply convolution operation to the input data followed by an activation function to produce the output. To improve training of CNN, Residual Neural Network (ResNet) were first introduced by He *et al.* . in (He et al. 2016). To understand ResNets, let us denote the underlying mapping between input (x) and output of network as $H(x)$. Then residual mapping can be defined as,

$$F(x) = H(x) - x. \tag{4}$$

Simply speaking, residual mapping is the difference between input and expected output of network. The original mapping can now be defined in terms of residual mapping as

$$H(x) = F(x) + x. \tag{5}$$

There is ample evidence in the literature indicating that network depth is of crucial importance and deeper networks in general achieve better results (Simonyan and Zisserman 2014; Szegedy et al. 2015). With ResNets it becomes easier to train big networks.

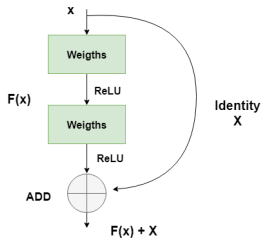


Figure 5: Residual Network (He et al. 2016)

3 Proposed sensing methodology by hardware constraints and deep learning based image restoration

Bearing the above hardware constraints in mind, in this paper, we propose a new framework for image sensing and using deep learning based technique to reconstruct the desired image.

3.1 Sensing methodology with hardware constraints

As mentioned before, a simple schematic of proposed sensing methodology is shown in Fig. 1. In this work, simulations using clean images is performed to mimic the proposed sensing methodology to verify the concept. It is expected that reconstruction performance will remain approximately the same in a real system.

The proposed sensing methodology consists of 6 stages. In the stage 1-3, image gets compressed using downsampling, bit truncation and JPEG. Stage 1-2 can be performed on the image sensor itself. Stage 3 happens on a JPEG chip or Digital Signal Processor (DSP). Stage 4 represents transmission of image which can be a wireless medium or an on-chip bus or even storage. Stage 5 can happen on a DSP processor in the device itself or it can be clubbed together with Stage 6 and can happen in the cloud or on the ML processor on the image acquisition device itself. The idea of this work is to save energy during acquisition *i.e.* , from stage 1 to stage 5 as these processes often consume a significant amount of power in edge devices and stage 6 is not required unless a user is viewing image (*e.g.* smart phones, surveillance cameras *etc.*) or a computer program is operating on images *e.g.* object recognition. For some applications like drone transmitting a surveillance footage to a base station power consumption in stage 6 is not an issue. For stage 6, we propose a DNN network for image reconstruction. The subsequent paragraphs describe each process in more detail.

In stage 1, image is downsampled using simple binning (averaging operation) of pixels. The number of pixels that are averaged depends on downscaling factor. As mentioned before, while this operation happens on raw image, this work simulates this process on clean images for the sake of simplicity. Further lossy compression is performed using bit truncation in stage 2. We perform the task of bit truncation in the following way

$$Truncated\ Pixel = round((averaged\ pixel) \times 2^{-N}),$$

where N is the number of bits to be truncated. In this work N is in the range $[0, 3]$. Since the truncated pixel does not get multiplied with 2^N after rounding operation, the image appears darker after truncation operation for $N > 0$. Simply speaking, we left-shift the pixel bits by the amount we want to truncate which makes the image appear dark. Thus the swing of the pixel values also gets reduced by a factor of 2^N. This makes the image more compressible using JPEG in stage 3 because there is more loss of LSB bits than MSB bits in JPEG compression. This also means that there is more lossy compression induced artefact. The JPEG quality is varied in the range [70,100] in our experiments. Currently, we assume perfect transmission of image in stage 4. When JPEG image is decompressed in stage 5, it gets multiplied by a factor of 2^N to restore the brightness. Because of bit truncation and lossy JPEG compression, artefact get introduced in JPEG decompressed image in addition to the loss of resolution (caused by downsampling via binning). Following this, our proposed LISR-net is employed to restore the desired high resolution image.

3.2 LISR-net

We propose the LISR-net, denoting Lossy Image Super-Resolution network, to finish the task in Stage 6, with architecture shown in Fig. 6. LISR-net is inspired by the orig-

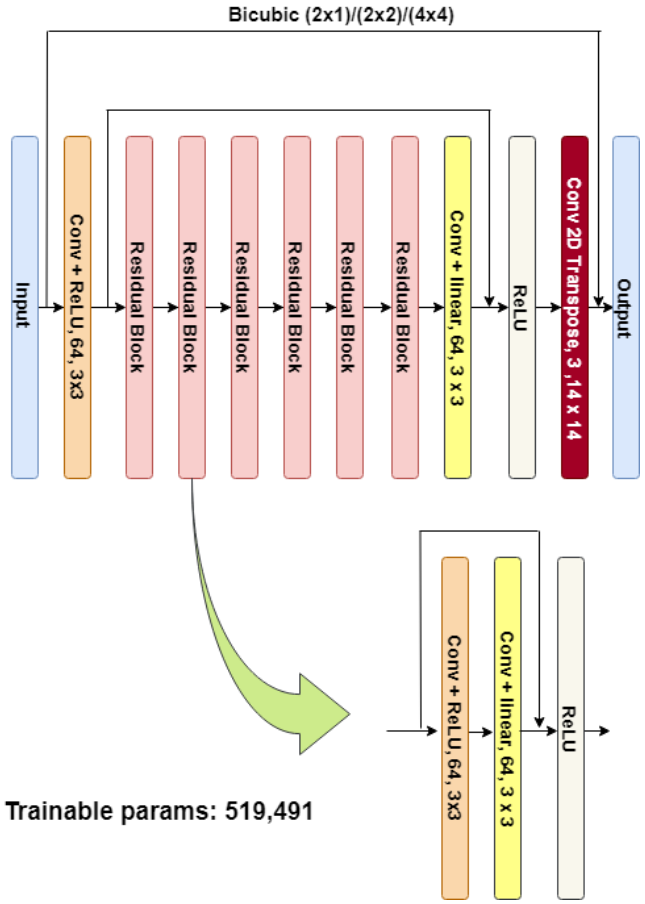


Figure 6: DNN network used in this work.

inal ResNet architecture (He et al. 2016), EDSR architecture (Lim et al. 2017) and SRCNN architecture (Ledig et al. 2017) with some differences. A comparison of ResNet block is shown in Fig. 7. To start, the basic ResNet block used in this work uses ReLU (Rectified Linear Unit) layer in the end like original ResNet network which allows more non-linearity and better performance. However it gets rid of Batch Normalization (BN) network as in EDSR Network because BN is not suitable for image super-resolution. Also, unlike EDSR, our LISR-net avoids learning a complete image; it only learns the residual between the bicubic interpolated image and the actual image. This is achieved by making a bypass connection between input and output using bicubic interpolation function as shown in Fig. 6. Since the residuals are mostly close to zero, the training is speeded up and the model complexity gets significantly reduced.

We train a separate network for a given downsampling factor, bit truncation and JPEG Quality factor to restore the image quality and resolution. Thus there are 48 different training tasks (4 cases of JPEG Quality, 4 cases of bit truncation and 3 cases of downsampling). The hyper-parameters are kept same for each training task.

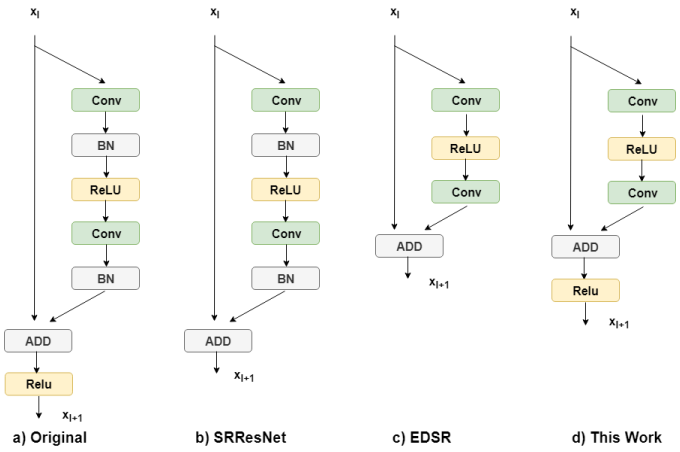


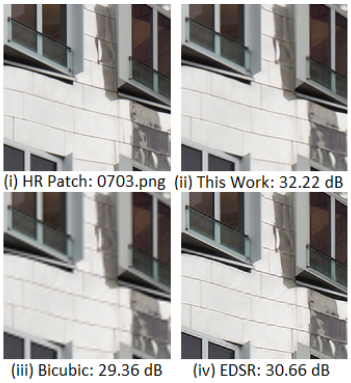
Figure 7: A comparison of basic ResNet blocks

4 Experimental results

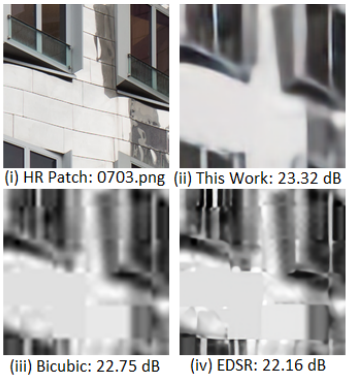
We use DIV2K dataset (Agustsson and Timofte 2017) for training and evaluation; DIV2K dataset is a high-quality (2K resolution) image dataset, which consists of 800 training images, 100 validation images and 100 testing images. Since the testing images are not made public, we use the last 100 images from training set (*i.e.* image 0701.png to 0800.png) as the testing set. The network uses patch size of 128×128 and $70,000$ training samples in a batch size of 64. The network is trained for 24 epochs.

Peak Signal to Noise Ratio (PSNR) is used as a metric to measure reconstruction performance with original high resolution image as the baseline. The results for reconstruction are shown in Table 1 for full sized testing dataset. One can see from the result that our LISR-net outperforms bicubic. For testing purposes, this work also predicted the output of EDSR network for the downsampled images used in our work. As expected the performance of the EDSR network falls sharply as it is not trained to handle the noise due to bit truncation, averaging and JPEG. One can also see that the performance of EDSR gets worse than bicubic as the image quality degrades. It also serves as a proof that our LISR-net does train itself properly to handle the degradation induced by JPEG, bit truncation and binning. Some samples of reconstructed images including best case (Q = 100 and B.W. = 8) and worst case (Q = 70 and B.W. = 5) are shown in Fig. 8. One can see that bicubic interpolated images are more noisy and less sharper than the images generated by LISR-net and EDSR.

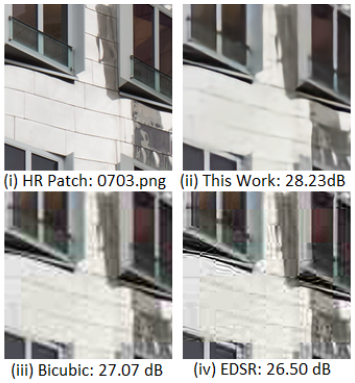
While the performance of the network proposed in this work might seems inferior to numbers reported in the EDSR paper (Lim et al. 2017), the focus in this work is to *perform on-sensor compression to reduce data traffic and save energy*. This is achieved by bit truncation and pixel binning both of which can be performed on commercially available image sensors. Almost all existing works of super-resolution use bicubic downsampling method which yields better image but cannot be performed on the image sensor. Our proposed LISR-net is also simpler than EDSR network and a comparison is shown in Table 2. Pixel binning and bit trun-



(a) 2×2 $S.R., Q = 100, B.W. = 8$

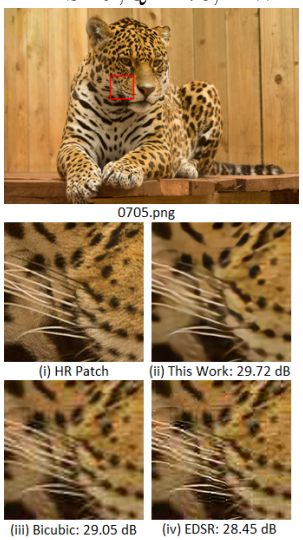


(b)4×4 $S.R., Q = 70, B.W. = 5$

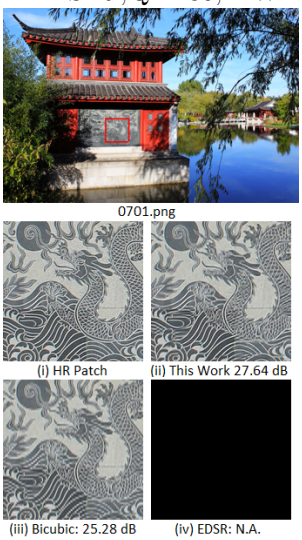


(c) 2×2 $S.R., Q = 80, B.W. = 6$

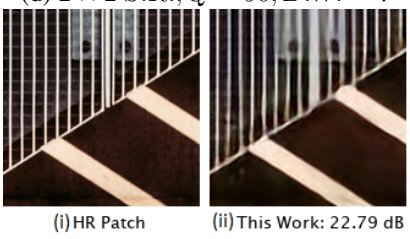
(d) 2×2 $S.R., Q = 90, B.W. = 7$

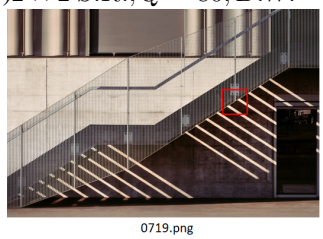


(e)2×2 $S.R., Q = 80, B.W. = 6$



(f)2×1 $S.R., Q = 100, B.W. = 8$





(g)4×4 $S.R., Q = 90, B.W. = 6$

Figure 8: Selected reconstructed images. For Fig. (f) EDSR result is not available as it is not designed for 2×1 super resolution.

Table 1: Reconstruction results for DIV2K dataset (0701.png-0800.png). PSNR metric in dB. Bicub. refers to bicubic interpolation, EDSR refers to the EDSR network in paper (Lim et al. 2017) and B.W. referes to the bitwidth of the image.

$Quality$	$B.W.$	$This\ Work$ 4×4	$BiCub.$ 4×4	$EDSR$ 4×4	$This\ Work$ 2×2	$BiCub.$ 2×2	$EDSR$ 2×2	$This\ Work$ 2×1	$BiCub.$ 2×1
100	8	**27.91**	26.75	27.28	**32.74**	30.97	31.61	**34.96**	33.19
100	7	**27.80**	26.68	27.12	**32.43**	30.78	31.14	**34.58**	32.86
100	6	**27.50**	26.44	26.59	**31.71**	30.09	30.04	**33.60**	31.96
100	5	**26.59**	25.72	25.26	**29.97**	28.76	27.87	**31.30**	29.98
90	8	**27.21**	26.36	26.29	**31.39**	30.18	29.88	**33.41**	32.12
90	7	**26.58**	25.87	25.51	**30.32**	29.35	28.69	**32.27**	31.11
90	6	**25.76**	25.12	24.63	**29.03**	28.14	27.38	**30.72**	29.64
90	5	**24.61**	24.05	23.48	**27.74**	26.49	25.75	**28.62**	27.63
80	8	**26.60**	25.90	25.57	**30.42**	29.44	28.82	**32.37**	31.24
80	7	**25.84**	25.23	24.80	**29.23**	28.38	27.68	**31.01**	29.98
80	6	**24.93**	24.37	23.92	**27.88**	27.06	26.43	**29.40**	28.40
80	5	**23.76**	23.20	22.69	**26.13**	25.36	24.76	**27.33**	26.37
70	8	**26.18**	25.55	25.15	**29.76**	28.88	28.20	**31.68**	30.60
70	7	**25.39**	24.80	24.38	**28.53**	27.73	27.09	**30.25**	29.24
70	6	**24.43**	23.87	23.44	**27.13**	26.35	25.78	**28.56**	27.58
70	5	**23.16**	22.60	22.11	**25.37**	24.58	24.03	**26.52**	25.50

Table 2: Comparison of networks.

Network	Residual Blocks	Trainable Weights
This Work	6	$.5M$
EDSR	32	$43M$

cation lead to significant reduction in raw data generated from image sensor. An analysis of compression of raw data is provided in Table 3. It is measured as follows

$$Raw\ Data\ Compression = \frac{8 \cdot N - B}{8 \cdot N}, \qquad (6)$$

where N represents number of pixels binned together and B represents bitwidth (B.W.) of pixel of downsampled image. One can achieve $50\% - 96\%$ reduction in raw data. Reduction in raw data means significant energy saving in downstream processing. One can achieve approximately proportional savings in energy for the same frequency of operation or one can employ DVFS to achieve quadratic scaling in energy reduction. For same frequency of operation, an estimation of power savings for different image sensor designs is shown in Table 4. One can see from the table that there is approximately proportional savings in power consumption. Apart from savings in power, the system also becomes faster as there is less data to process. Reduction in bitwidth can also result in exponential reduction in power consumed at ADC (Section 2.1). Additionally, it can lead to more than proportional savings in energy in image processing circuits as LSB's switch more fro1m one pixel to another than MSB in an image.

Table 3: Raw data Compression Results. B.W. refers to the bitwidth of image. Raw Compression does not depend on JPEG Quality factor Q.

	B.W. = 8	B.W. = 7	B.W. = 6	B.W. = 5
2×1	50%	56.25%	62.5%	68.75%
2×2	75%	78.12%	81.25%	84.37%
4×4	93.75%	94.53%	95.31%	96.09%

Let us assume that image is being readout to an 8-bit wide data bus in column-wise fashion for each color channel. A table of switching activity measurement for such a case is shown in Table 5. This will lead to significant savings in dynamic power consumption as explained earlier in Sec. 2.2. Reduction in raw data also leads to reduction in processed image size after JPEG compression. The results for this are shown in Table 6, which shows the size of the resulting image as a percentage of the size of the original high resolution image stored in lossless JPEG format. One can see that com-

Table 4: Power Savings Estimation. D1 and D2 refers to image sensor design in Ref. (Oike and El Gamal 2013) and (Shin et al. 2012) respectively. JPEG design is from Ref. (Shichao, Zhizhong, and Xin 2016). C.R. stands for Raw Data Compression Ratio and Opr. stands for Operation. Data has been normalized for 120 fps operation.

	D1 (mW)	**D2 (mW)**	**D1 (mW)**	**D2 (mW)**	**D1 (mW)**	**D2 (mW)**
C.R. / Opr.	0%	0%	50%	50%	96.1%	96.1%
I/O	27	70	13.5	35	1.05	2.73
ADC	60	209	30	104.5	2.34	8.15
Pixel	1.8	23	1.8	23	1.8	23
Other	4.2	20	4.2	20	4.2	20
JPEG	13.18	386.3	6.59	193.15	.51	15.06
Total	106.18	708.3	56.09	375.65	9.9	68.94
Savings	0%	0%	**47.18%**	**46.97%**	**90.67%**	**90.27%**

Table 5: Switching activity analysis of images. For DIV2K dataset (0701.png-0800.png).

$Bit\ Position$	$Swiching\ Activity\ (\alpha)$
$0 = LSB$	0.48
1	0.46
2	0.41
3	0.33
4	0.25
5	0.17
6	0.09
$7 = MSB$	0.04

pressed image size ranges from 22.7% to less than 1% of the size of lossless image.

As mentioned before, this work does not take into account the energy spent in reconstruction of the image as the aim is to reduce the energy for acquisition. The images can be reconstructed either on edge devices or on cloud, and an example of edge device can be SmartTV. One can reduce the image/video data transmission bandwidth. The image/video can then be reconstructed using the GPU available in smart TVs. For the case of smartphones, image can be acquired in low resolution mode and can be reconstructed back in the cloud. Since the images and photos are generally uploaded to cloud storage nowadays, reconstruction in cloud is technically feasible. For the case of drones transmitting video surveillance footage, the compression can reduce the power consumption in drone. It can also make transmission feasible in noisy environment or over longer distance because small image size offers an opportunity to aggressively encode the message packet with ECC.

Table 6: Size Comparison. For DIV2K dataset (0701.png-0800.png). Measured as percentage with respect to original image in lossless JPEG format.

$Quality$	$Bitwidth$	2×1	2×2	4×4
100	8	22.7%	12.27%	3.48%
100	7	17.38%	9.41%	2.66%
100	6	12.88%	6.95%	1.92%
100	5	9.41%	5.11%	1.41%
90	8	7.77%	4.29%	1.21%
90	7	5.32%	3.07%	0.84%
90	6	3.68%	1.92%	0.57%
90	5	2.45%	1.27%	0.39%
80	8	5.32%	2.86%	0.82%
80	7	3.48%	1.88%	0.55%
80	6	2.25%	1.23%	0.37%
80	5	0.78%	0.80%	0.25%
70	8	4.09%	2.25%	0.65%
70	7	2.86%	1.47%	0.43%
70	6	1.76%	0.96%	0.29%
70	5	1.17%	0.61%	0.18%

5 Conclusion and future work

This paper establishes the use of DNNs for energy savings in process of image acquisition considering hardware constraints. For a given image quality requirement, one can acquire a low resolution image to save power and augment resolution and quality using DNN network in a fashion discussed in this paper. This would reduce the effort spent on the process of image acquisition resulting in improvement in power and performance parameters of the imaging device. The proposed methodology makes the system programmable *i.e.* , user can shift from low resolution image acquisition to traditional high resolution image acquisition through software control of binning operation in imaging systems which is generally exposed to the system programmer. Using these techniques, one can achieve more than 50% reduction in raw data and at least similar reduction in power while maintaining the PSNR above 30 dB.

In the future, we would like to use a more accurate simulation for an imaging system. Deeper networks can also be studied to improve the reconstruction performance. We would also like to explore the prediction network using integer operation instead of floating point to make it suitable for edge devices and real time operation.

References

3GPP. 2017. Technical Specification Group Radio Access Network; NR; Multiplexing and channel coding. Techni-

cal Specification (TS) 38.212, 3rd Generation Partnership Project (3GPP). Version 15.0.0.

Agustsson, E., and Timofte, R. 2017. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Cho, J.; Choi, J.; Kim, S.-J.; Shin, J.; Park, S.; Kim, J. D.; and Yoon, E. 2013. A 5.9 μm-pixel 2d/3d image sensor with background suppression over 100klx. In *VLSI Circuits (VLSIC), 2013 Symposium on*, C6–C7. IEEE.

Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2016. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence* 38(2):295–307.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Huang, H.-Y.; Conge, P. A.; and Huang, L.-W. 2011. Cmos image sensor binning circuit for low-light imaging. In *2011 IEEE Symposium on Industrial Electronics and Applications*, 586–589. IEEE.

Kim, J.; Kwon Lee, J.; and Mu Lee, K. 2016. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1646–1654.

Kumagai, O.; Niwa, A.; Hanzawa, K.; Kato, H.; Futami, S.; Ohyama, T.; Imoto, T.; Nakamizo, M.; Murakami, H.; Nishino, T.; et al. 2018. A 1/4-inch 3.9 mpixel low-power event-driven back-illuminated stacked cmos image sensor. In *2018 IEEE International Solid-State Circuits Conference-(ISSCC)*, 86–88. IEEE.

Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint*.

Lim, B.; Son, S.; Kim, H.; Nah, S.; and Mu Lee, K. 2017. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 136–144.

Liu, J.; Sun, W.; Su, Y.; Jing, P.; and Yang, X. 2019. Be-calf: bit-depth enhancement by concatenating all level features of dnn. *IEEE Transactions on Image Processing*.

Liu, J.; Sun, W.; and Liu, Y. 2017. Bit-depth enhancement via convolutional neural network. In *International Forum on Digital TV and Wireless Multimedia Communications*, 255–264. Springer.

Luo, Y.; Ghose, S.; Cai, Y.; Haratsch, E. F.; and Mutlu, O. 2018. Heatwatch: improving 3d nand flash memory device reliability by exploiting self-recovery and temperature awareness. In *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 504–517. IEEE.

Mao, X.; Shen, C.; and Yang, Y.-B. 2016. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in neural information processing systems*, 2802–2810.

Oike, Y., and El Gamal, A. 2013. Cmos image sensor with per-column $\sigma\delta$ adc and programmable compressed sensing. *IEEE Journal of Solid-State Circuits* 48(1):318–328.

Shichao, Y.; Zhizhong, H.; and Xin, C. 2016. A scalable multi-pipeline jpeg encoding architecture. In *2016 28th International Conference on Microelectronics (ICM)*, 369–372. IEEE.

Shin, M.-S.; Kim, J.-B.; Kim, M.-K.; Jo, Y.-R.; and Kwon, O.-K. 2012. A 1.92-megapixel cmos image sensor with column-parallel low-power and area-efficient sa-adcs. *IEEE Transactions on Electron Devices* 59(6):1693–1700.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Su, Y.; Sun, W.; Liu, J.; Zhai, G.; and Jing, P. 2019. Photo-realistic image bit-depth enhancement via residual transposed convolutional neural network. *Neurocomputing*.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Timofte, R.; Gu, S.; Wu, J.; and Van Gool, L. 2018. Ntire 2018 challenge on single image super-resolution: methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 852–863.

Timofte, R.; De Smet, V.; and Van Gool, L. 2014. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Asian conference on computer vision*, 111–126. Springer.

Umeda, S.; Watanabe, H.; Ikai, T.; Hashimoto, T.; Chujoh, T.; and Ito, N. 2019. Joint super-resolution and bit depth extension by dnn. In *International Workshop on Advanced Image Technology (IWAIT) 2019*, volume 11049, 1104925. International Society for Optics and Photonics.

Wang, Z.; Chen, J.; and Hoi, S. C. 2019. Deep learning for image super-resolution: A survey. *arXiv preprint arXiv:1902.06068*.

Yang, W.; Zhang, X.; Tian, Y.; Wang, W.; and Xue, J.-H. 2018. Deep learning for single image super-resolution: A brief review. *arXiv preprint arXiv:1808.03344*.

Yip, M., and Chandrakasan, A. P. 2013. A resolution-reconfigurable 5-to-10-bit 0.4-to-1 v power scalable sar adc for sensor applications. *IEEE Journal of Solid-State Circuits* 48(6):1453–1464.

Yoshihara, S.; Nitta, Y.; Kikuchi, M.; Koseki, K.; Ito, Y.; Inada, Y.; Kuramochi, S.; Wakabayashi, H.; Okano, M.; Kuriyama, H.; et al. 2006. A 1/1.8-inch 6.4 mpixel 60 frames/s cmos image sensor with seamless mode change. *IEEE Journal of Solid-State Circuits* 41(12):2998–3006.

Yu, K.; Dong, C.; Loy, C. C.; and Tang, X. 2016. Deep convolution networks for compression artifacts reduction. *arXiv preprint arXiv:1608.02778*.

Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; and Zhang, L. 2017. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing* 26(7):3142–3155.

Zhou, Z.; Pain, B.; and Fossum, E. R. 1997. Frame-transfer cmos active pixel sensor with pixel binning. *IEEE Transactions on electron devices* 44(10):1764–1768.