Supplemental Material

Here, we provide additional details of our proposed method, including a detailed description of the dataset, architectural design details, FLOPs, visualizations of the differences in sub-pixel and noise distribution, feature visualization maps, additional visual results, explanations of differences compared to similarly named methods.

A. Detailed Description of the Dataset

Here, we provide a comprehensive description of the dataset used in our method.

SyntheticBurst Dataset. The SyntheticBurst dataset comprises 46,839 bursts for training and 300 bursts for testing. Each burst consists of 14 low-resolution (LR) RAW images synthesized from a single sRGB image captured by a Canon camera. The process begins with converting the sRGB image to linear RGB values using an inverse camera pipeline. Random translations and rotations are then applied to generate a shifted burst. The transformed images are subsequently downsampled using a bilinear kernel and noise is added to obtain the low-resolution and noisy burst. Finally, Bayer mosaicking is applied to produce the input RAW burst.

BurstSR Dataset. The BurstSR dataset includes 5,405 patches for training and 882 patches for testing. The LR RAW images and the corresponding high-resolution (HR) sRGB image are captured using a Samsung smartphone camera and a Canon DSLR camera, respectively.

RealBSR-RAW Dataset. The RealBSR-RAW dataset comprises 579 groups of burst images for a scale factor of 4. Each group includes 14 burst LR images and a ground truth (GT) HR image. The images are captured using a Sony DSLR camera (Alpha 7R), where a sequence of 14 LR images is taken by pressing the camera shutter and optically zooming the camera to capture an HR image. These images are collected across various scenes including buildings (museums, churches, office buildings, towers, etc.), posters, plants/trees, sculptures, and ships. The dataset contains 21 indoor groups and 618 outdoor groups. Due to the different fields of view between LR and HR images, SIFT is used to crop LR sequences with reference to the collected HR counterpart. Considering that the distortion of RAW images is not corrected by the camera, their center regions are cropped into the RealBSR-RAW dataset. For model training, inputs are cropped into 160×160 patches, yielding 20,842 groups of paired patches for training and 2,377 groups for testing.

RealBSR-RGB Dataset. The RealBSR-RGB dataset consists of 639 groups of burst images for a scale factor of 4. This dataset is similar to the RAW version but involves processing by the camera ISP, necessitating color and luminance corrections between LRs and their HR counterparts. Similar to the RealBSR-RAW, inputs are cropped into 160×160 patches, resulting in 19,509 groups of paired patches for training and 2,224 groups for testing.

B. Architectural Details

This section delves into the architectural design details of our proposed method. Figure S1(a) illustrates the specific implementation details of our QSSM. In Figure S1(b), a schematic of the four scanning lines in the mamba component of our method is presented. The detailed implementation of the SE block within QSSM is shown in Figure S1(c). Figure S1(d) presents the specific implementation of our proposed MSFM.

C. FLOPs, and Running Time

Here, we provide a comparison of the FLOPs and running time between existing methods and our proposed approach. All performance metrics are measured with an input size of 14×4×48×48.

Due to the lack of open-sourced implementations for many methods in the table, we only included those for which we could gather statistics, and we once again presented the performance of existing methods to demonstrate the effectiveness of our method.

D. Visualizations of Sub-Pixel and Noise Distribution Differences

This section includes visualizations that demonstrate the differences in sub-pixel and noise distribution. After aligning the current frames with the base frame, we compute the residual images by subtracting each aligned pair, effectively highlighting the differences between them.

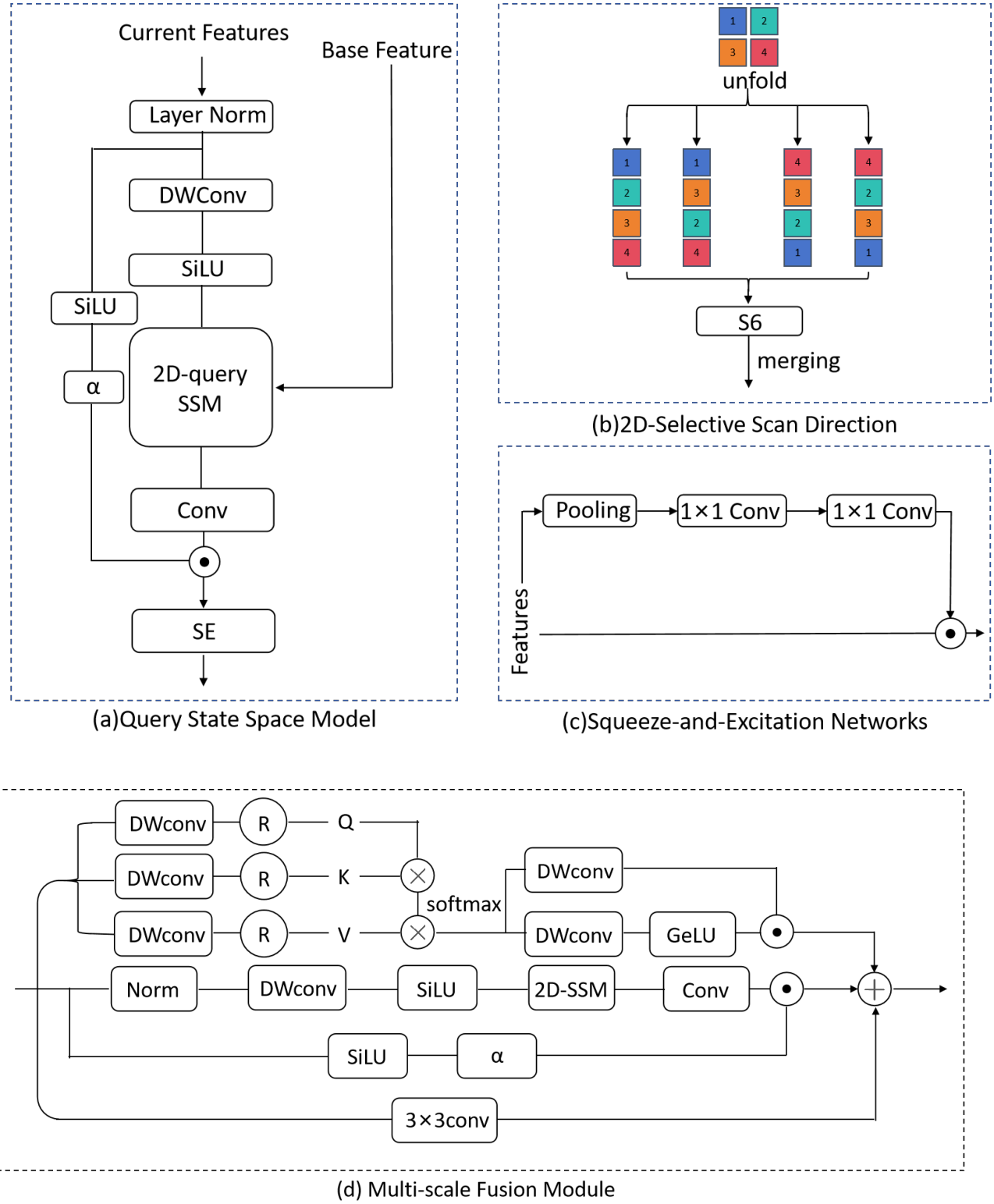

Figure S1:The implementation details of our proposed QSSM (Query State Space Model) and MSFM (Multi-scale Feature Merge), along with their specific internal mechanisms.

	Bicubic	High-ResNet	DBSR	MFIR	BIPNet	AFCNet	FBAnet	GMTNet	RBSR	Burstormer	**Ours**
PSNR↑	36.17	37.45	40.76	41.56	41.93	42.21	42.23	42.36	42.44	42.83	**43.12**
SSIM↑	0.91	0.92	0.96	0.96	0.96	0.96	0.97	0.96	0.97	0.97	**0.97**
FLOPs(G)	-	400	118	110	300	-	-	157	156	61.79	56.16
Running Time(ms)	-	46.3	54.67	420	130	-	-	200	119.93	40.86	42.1

Table S1:Comparison of PSNR, SSIM, Running Time, and FLOPs across different methods.

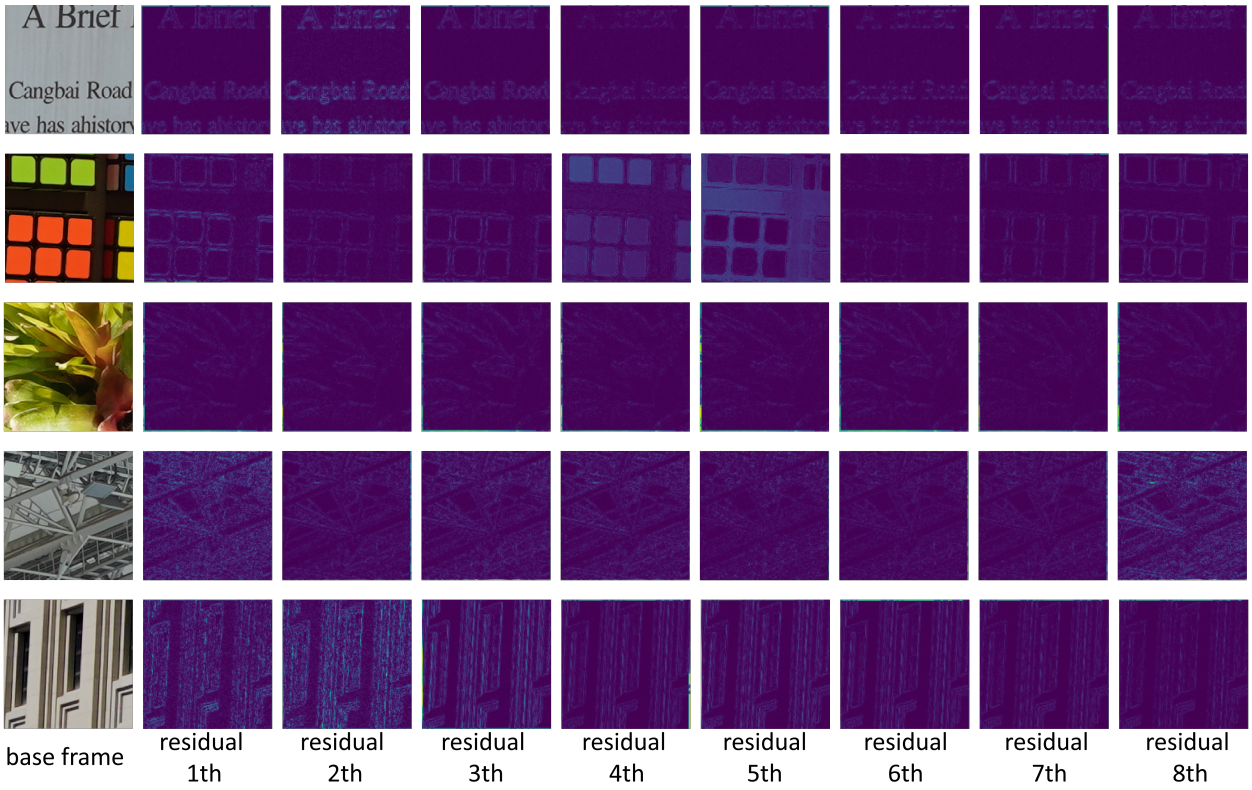

Figure S2:Residual images obtained by subtracting each of the 8 aligned current frames from the base frame.

As shown in Figure S2, it can be observed that the distribution of sub-pixels across multiple residual images shows consistency and often manifests as high-frequency signals, appearing in regions with significant gradients. In contrast, the distribution of high-frequency noise is random both spatially and across multiple frames. The QSSM leverages these observed characteristics of noise and sub-pixels to extract sub-pixel information that matches the content of the base frame from multiple current frames and utilizes joint multi-frame processing to remove random noise

E. Feature Visualization

Here, we present visualizations of the input and output features of our proposed QSSM and AdaUp modules. Features are shown in Figure S3.

The input and output features of the QSSM module, as depicted in the figures, reveal that after processing through the QSSM module, there is a noticeable reduction in noise within the features. This is achieved by leveraging the consistency of sub-pixels and the differential distribution of noise, effectively denoising through multi-frame integration. Additionally, due to the wide receptive field of the QSSM, the output contains more detailed features compared to the input. The QSSM is capable of extracting sub-pixel information from current features that match the content of the base feature, which is evident as the darker areas are more concentrated in specific structures or textures rather than being dispersed or uniformly distributed

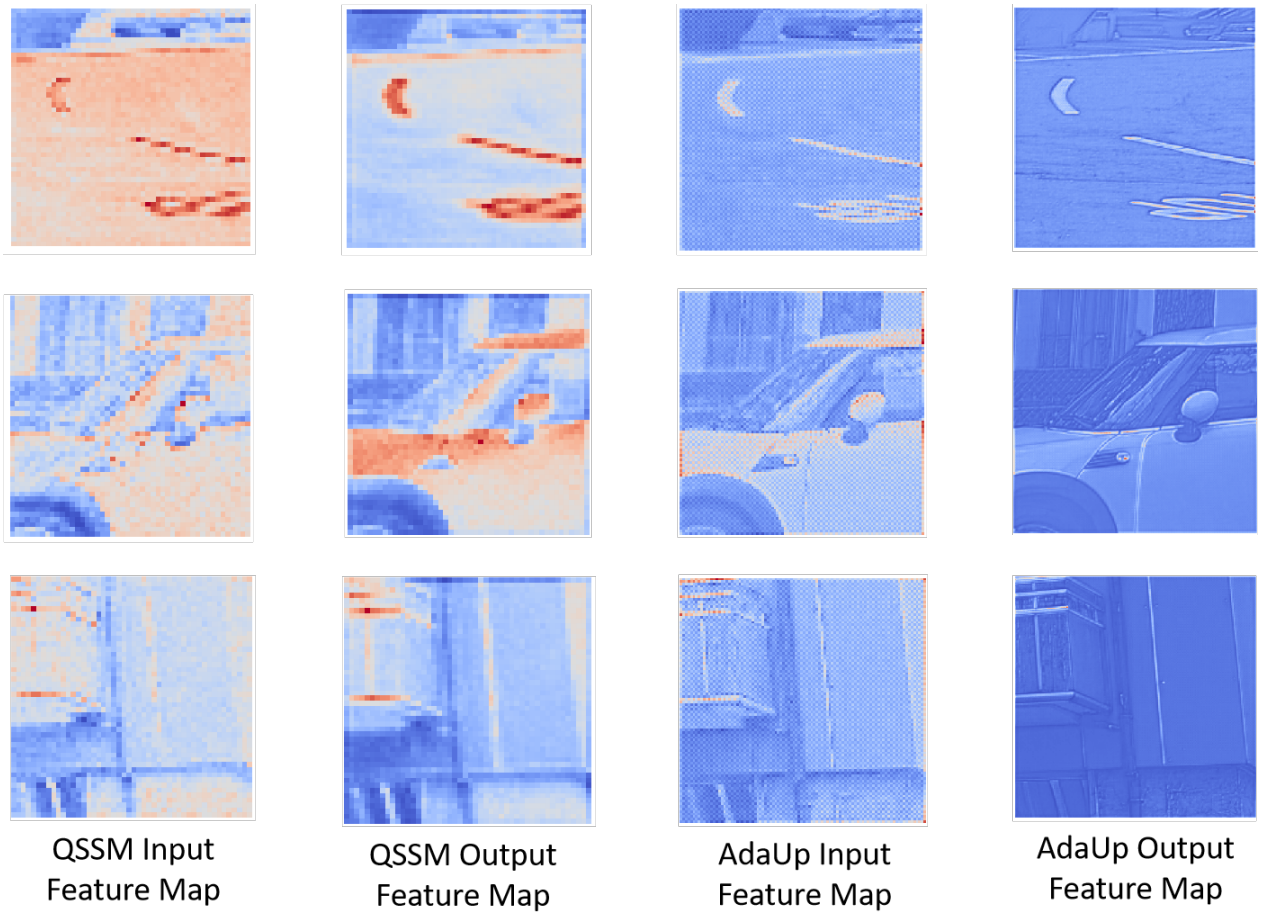

Figure S3:Input and output features of our proposed QSSM and AdaUp modules

globally. The features thus underscore the rationality of our proposed QSSM approach.

For AdaUp, the output clearly contains richer details compared to the input, while also eliminating the mosaic effect of Bayer RAW images. This enhancement is due to AdaUp's ability to adaptively adjust the weights of the upsampling kernels based on the energy distribution differences across different frames and channels within the same frame, allocating these kernels to various spatial positions. This characteristic of AdaUp allows for more effective utilization of multi-frame sub-pixel information, leading to higher-quality high-resolution image reconstruction.

F. Additional Visual Results

Figure S4, and Figure S5 show qualitative results of competing approaches on examples from the SyntheticBurst and RealBSR-RGB for 4× SR. The reproductions of our QMambaBSR are more detailed, sharper than those produced by the other methods.

G. Analysis and Discussion of Related Work

Although the title of this paper and QSSM are similar to [55], it is important to note that we only adopted a similar naming convention, and there are fundamental differences in both implementation methods and underlying principles. The most crucial aspect of QSSM is the modification of the SSM component. We designed it as a 2D-query SSM, which leverages the differences in sub-pixel and noise distribution through a gating mechanism to achieve clean sub-pixel extraction, specifically catering to low-level vision tasks like super-resolution. In contrast, [55] still employs the original SSM, which is designed to serve high-level vision tasks.

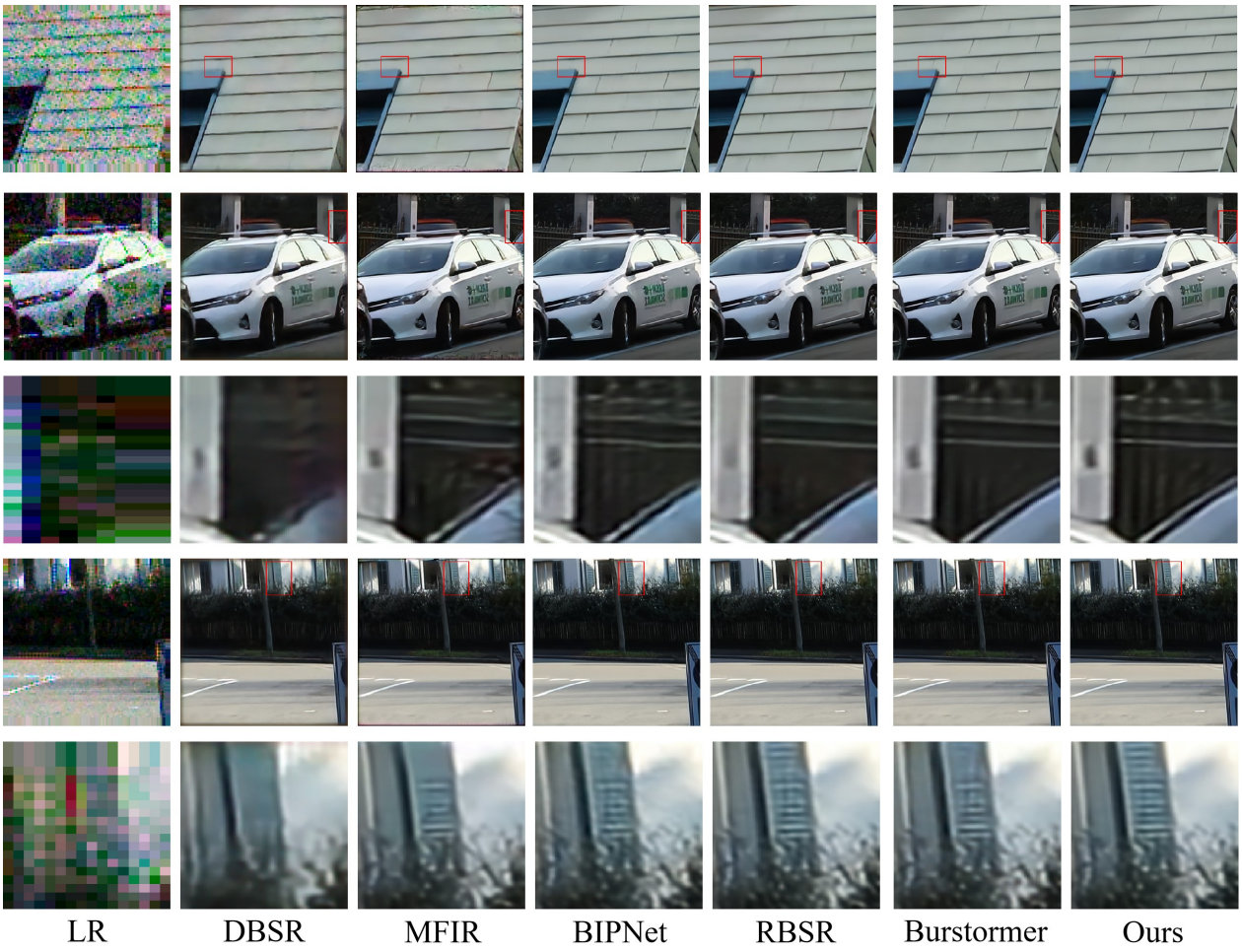

Figure S4 : Burst super-resolution (4×) results on SyntheticBurst dataset

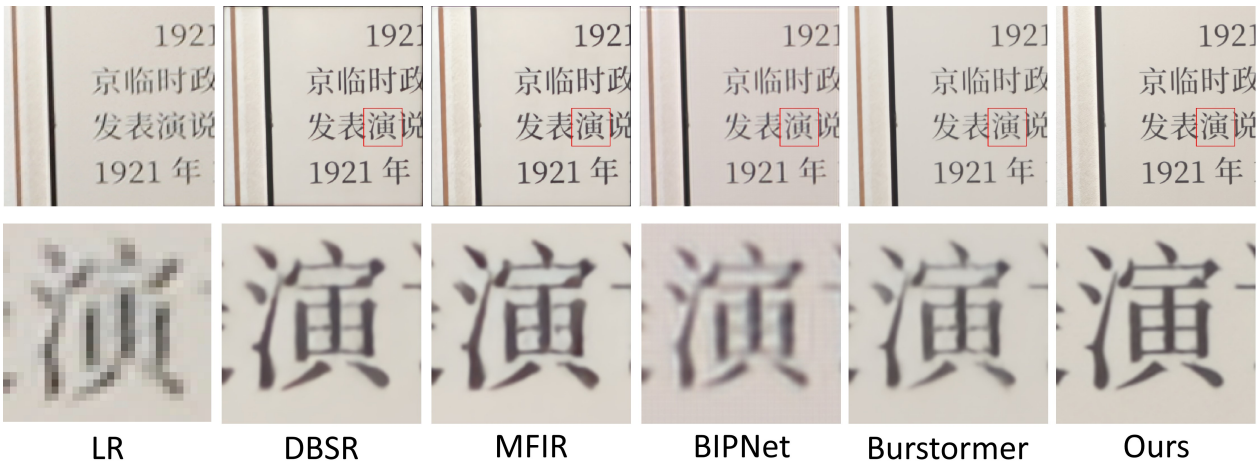

Figure S5 : Burst super-resolution (4×) results on RealBSR-RGB dataset