Supplementary Materials for
“Memory-Efficient Network for Large-scale Video Compressive Sensing”

Ziheng Cheng, Bo Chen*, Guanliang Liu, Hao Zhang, Ruiying Lu and Zhengjue Wang
Xidian University
zhcheng@stu.xidian.edu.cn, bchen@xidian.edu.cn
{lgl_xidian,zhanghao_xidian,ruiyinglu_xidian,zhengjuewang}@163.com

Xin Yuan*
Bell Labs
xyuan@bell-labs.com

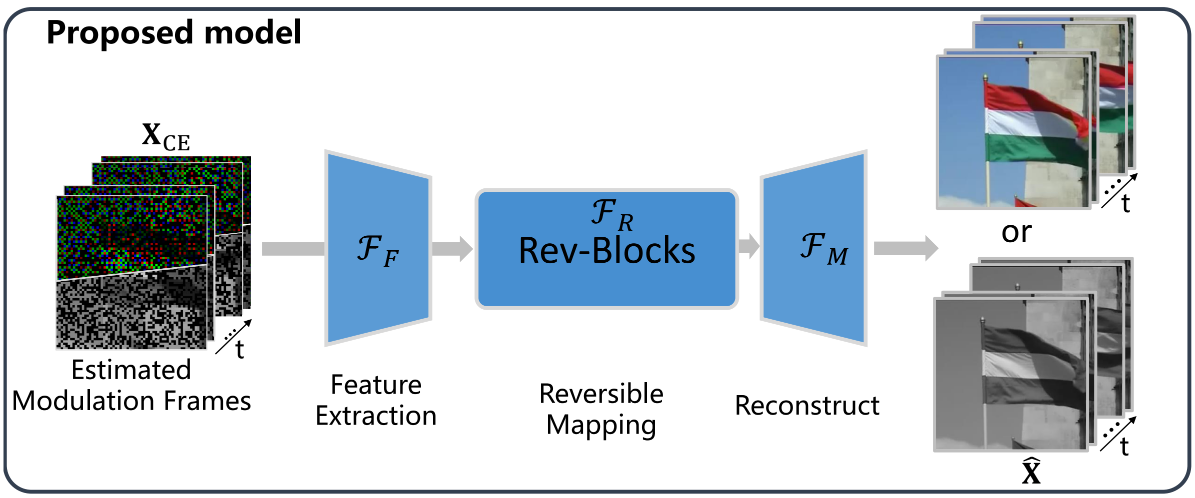

Figure 1: The overview of the proposed model RevSCI-net. Given the estimated modulation frames which are generated from the original measurement and masks (left), RevSCI-net (middle) extracts the high-dimensional feature and fuses them to obtain the desired reconstruction videos (right).

Here, we present more details about the proposed model RevSCI-net and full results that could not be contained in the main paper due to the page limit.

1. Details for RevSCI-net

RevSCI-net contains three modules:

1) Feature extraction $\mathcal{F}_F$ uses four 3D CNN layers to capture the high-dimensional features of the input.

2) Feature level nonlinear mapping employs L reversible blocks $\mathcal{F}_R$ to transform the input features into the desired reconstruction domain.

3) Reconstruction $\mathcal{F}_M$ uses four 3D CNN layers to integrate the features to reconstruct the final video.

In Fig. 1, we again show the model overview and in this section, we present the details for each layer and hyperparameters in Table 1.

In Table 1, we present the full setting for $\mathcal{F}_R$ and $\mathcal{F}_M$. The structure of $\mathcal{F}_R$ depends on the number of reversible blocks L and the number of the group m, the layers setting of each block in $\mathcal{F}_R$ are the same *i.e.* two 3D convolutional layers and the channel number depending on the m. c_1 is the channel number of the feature after $\mathcal{F}_F$, and change c_1 to maintain the RevSCI-net with similar parameters in different groups (we set $c_1 = 64$ when $m = 2$). Note that, the differences between grayscale and color RevSCI-net are: 1) The input size. For grayscale the size is $1 \times B \times n_x \times n_y$, and for color reconstruction the size is $4 \times B \times \frac{n_x}{2} \times \frac{n_y}{2}$; 2) The stride of the last layer in $\mathcal{F}_F$ for color reconstruction is

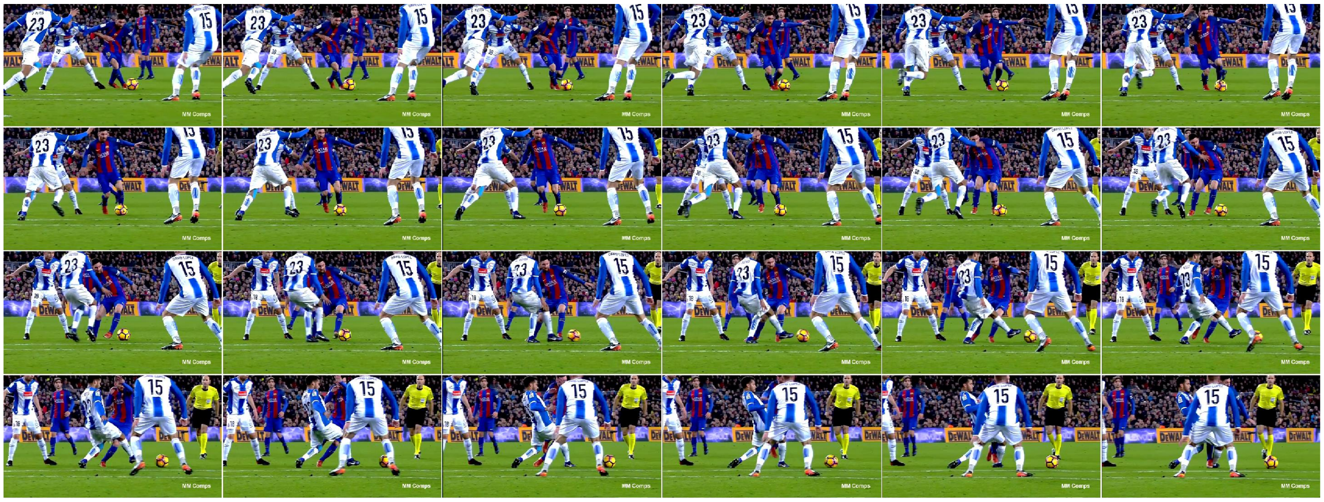

Figure 2: Ground truth for `Messi`.

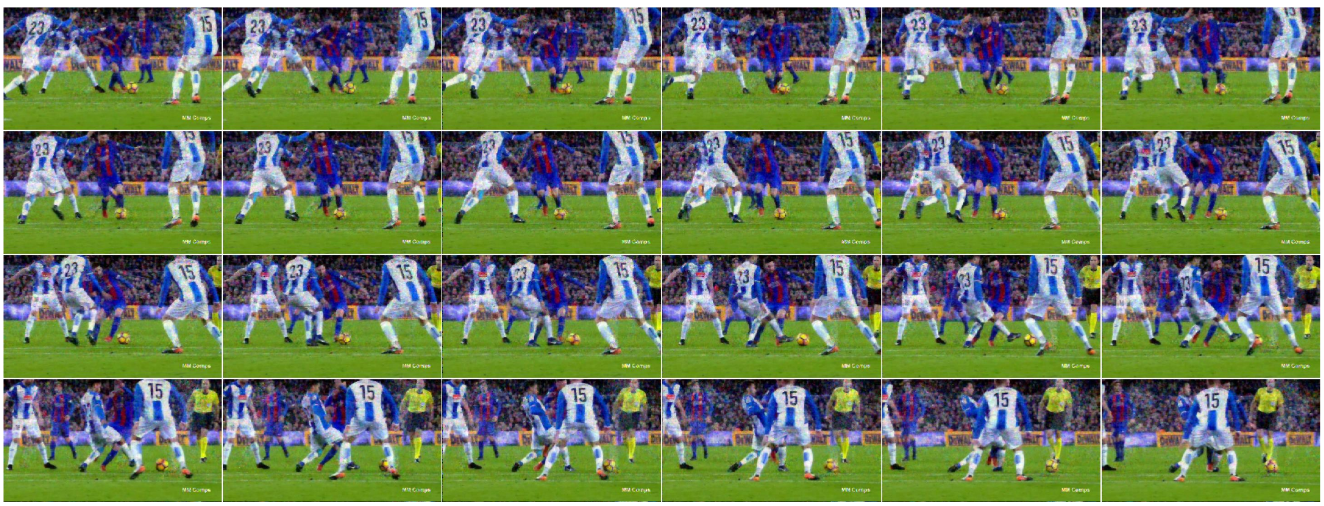

Figure 3: Reconstructed results by GAP-TV for `Messi`, PSNR: 18.56.

Table 1: Network architecture for the grayscale RevSCI-net.

Module	Operation	Kernel	Stride	Output Size ($c \times t \times h \times w$)
$\mathcal{F}_F$	Conv.	$5 \times 5 \times 5$	(1, 1, 1)	$16 \times B \times n_x \times n_y$
	Conv.	$3 \times 3 \times 3$	(1, 1, 1)	$32 \times B \times n_x \times n_y$
	Conv.	$1 \times 1 \times 1$	(1, 1, 1)	$32 \times B \times n_x \times n_y$
	Conv.	$3 \times 3 \times 3$	(1, 2, 2)	$c_1 \times B \times \frac{n_x}{2} \times \frac{n_y}{2}$
$\mathcal{F}_m$ **in** $\mathcal{F}_R$	Conv.	$3 \times 3 \times 3$	(1, 1, 1)	$\frac{c_1}{m} \times B \times \frac{n_x}{2} \times \frac{n_y}{2}$
	Conv.	$3 \times 3 \times 3$	(1, 1, 1)	$\frac{c_1}{m} \times B \times \frac{n_x}{2} \times \frac{n_y}{2}$
$\mathcal{F}_M$	Conv. Transpose	$3 \times 3 \times 3$	(1, 2, 2)	$32 \times B \times n_x \times n_y$
	Conv.	$3 \times 3 \times 3$	(1, 1, 1)	$32 \times B \times n_x \times n_y$
	Conv.	$1 \times 1 \times 1$	(1, 1, 1)	$16 \times B \times n_x \times n_y$
	Conv.	$3 \times 3 \times 3$	(1, 1, 1)	$1 \times B \times n_x \times n_y$

1 to keep the same size as input; 3) The output size of $\mathcal{F}_M$ is $3 \times B \times n_x \times n_y$ because the color (usually RGB) image is 3 channels.

2. More Experimental Results

We show additional experimental results of the proposed framework on the color large-scale simulated data `Messi` ($1080 \times 1920 \times 3 \times 24$) [1]. and real data `Wheel` ($256 \times 256 \times 14$), `Domino`, and `Water Balloon` ($512 \times 512 \times 50$) captured by SCI cameras. Note that, to our best knowledge, the proposed RevSCI-net is the first end-to-end training network (joint reconstruction and demosaicing) to recover such a large SCI scene (for the scale at the `Messi` data).

For better comparing the difference of these methods, we

[1]In order to generate the Bayer measurement, different from PnP-FFDNet we first select one pixel (corresponding to the Bayer filter) from three pixels (RGB) of the original video, and then they are modulated by the masks and are integrated into the Bayer measurement. Therefore the spatial resolution of the measurement is the same as the original video, instead of scaling it up by 2 times in the PnP-FFDNet paper [1]

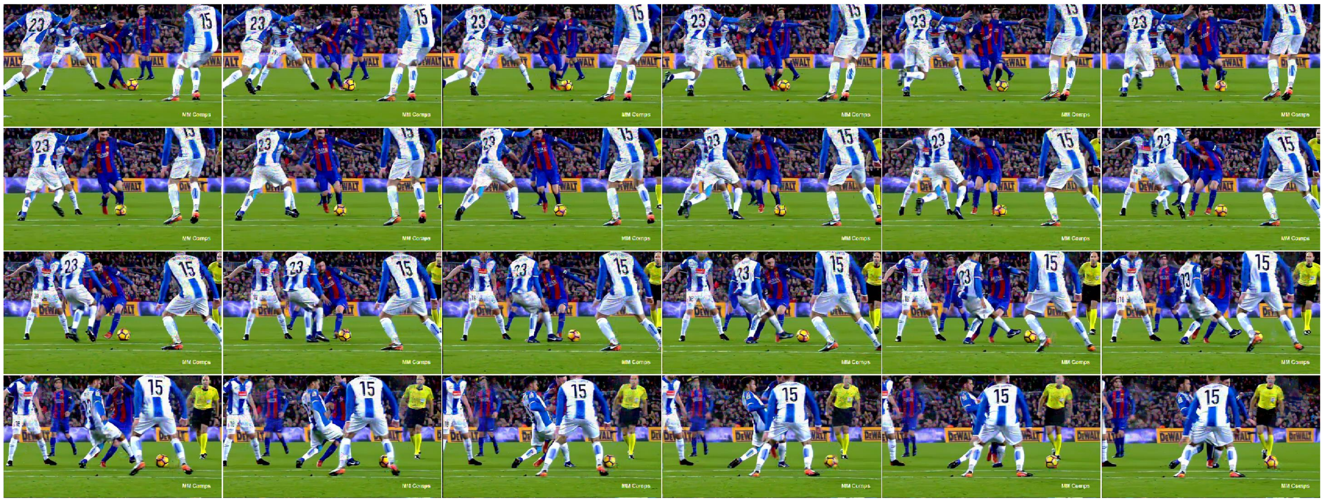

Figure 4: Reconstructed results by PnP-FFDNet for `Messi`, PSNR: 21.54.

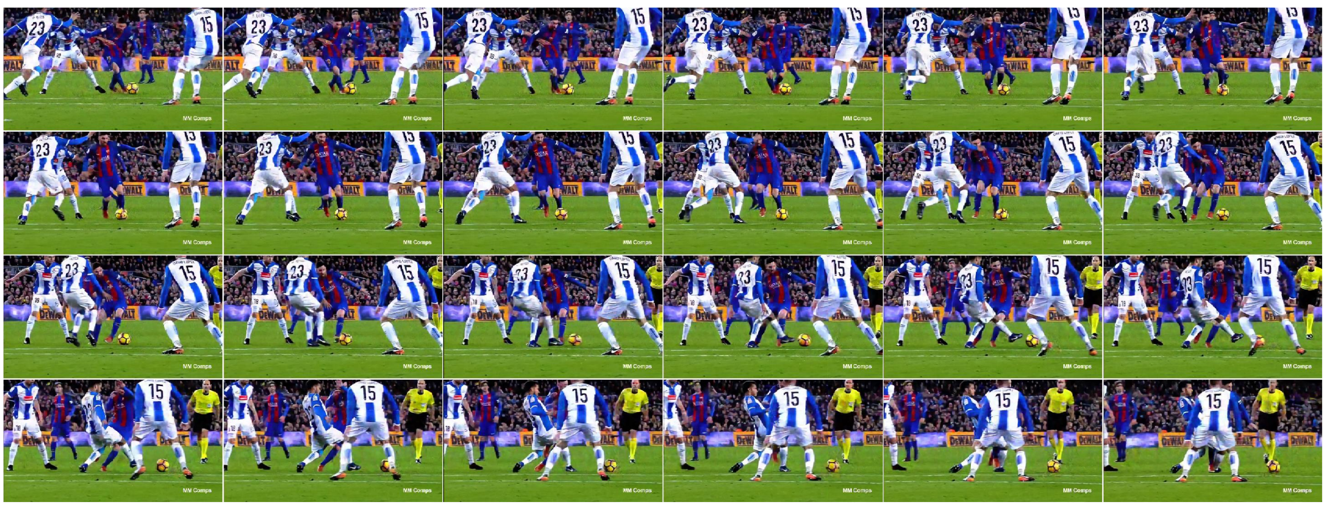

Figure 5: Reconstructed results by RevSCI-net for `Messi`, PSNR: 24.35.

also provide the videos in GIF format in the Supplementary Materials. As a further illustration of the results shown in the main paper, we show the whole reconstructed frames of `Messi` using different methods in Fig. 2 to Fig. 5. The reconstructed videos using different methods on the real data `Wheel`, `Domino`, and `Water Balloon` are shown in Fig. 6 to Fig. 14.

References

[1] Xin Yuan, Yang Liu, Jinli Suo, and Qionghai Dai. Plug-and-play algorithms for large-scale snapshot compressive imaging. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

Figure 6: The results of different methods on real data `Wheel`.

Figure 7: The results of GAP-TV on real data `Domino`.

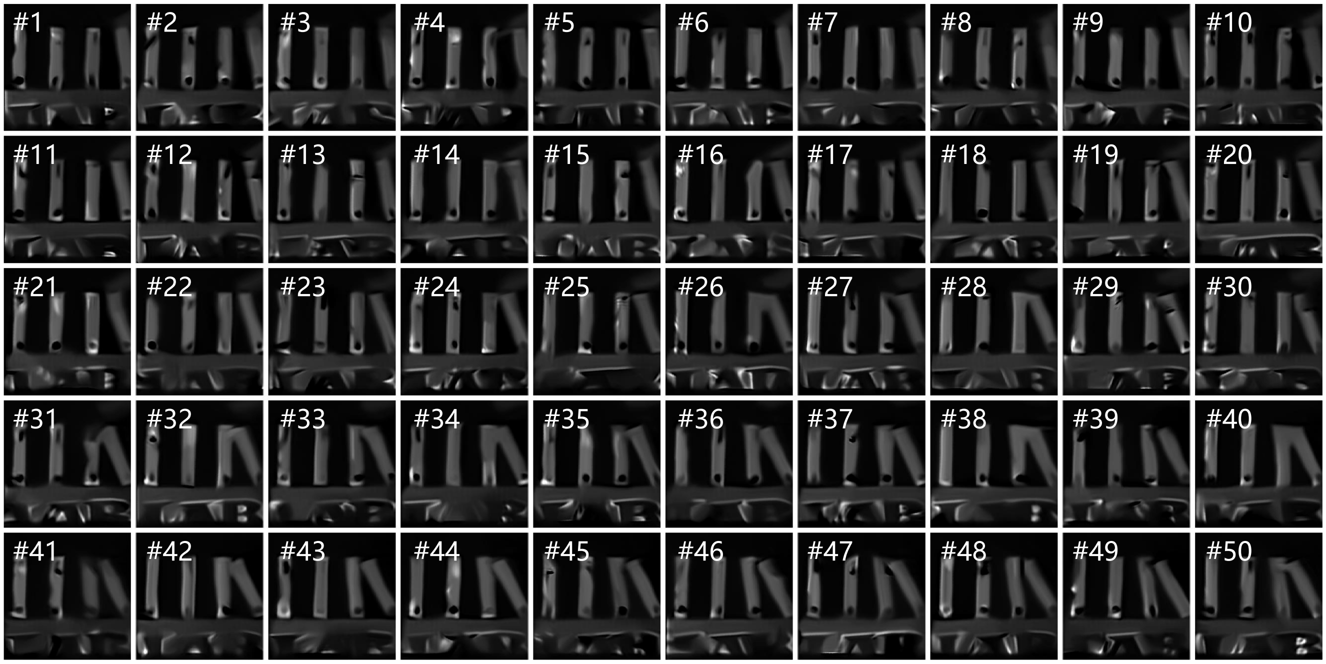

Figure 8: The results of DeSCI on real data `Domino`.

Figure 9: The results of PnP-FFDNet on real data `Domino`.

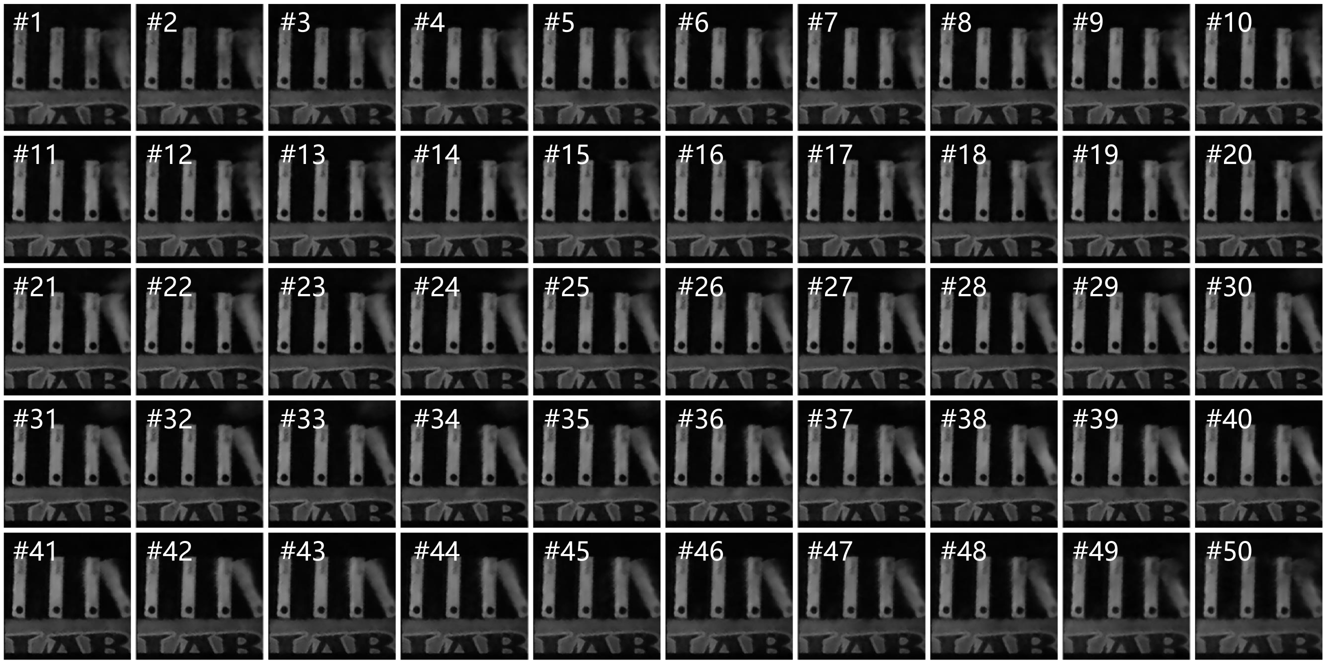

Figure 10: The results of RevSCI-net on real data `Domino`.

Figure 11: The results of GAP-TV on real data `Water Balloon`.

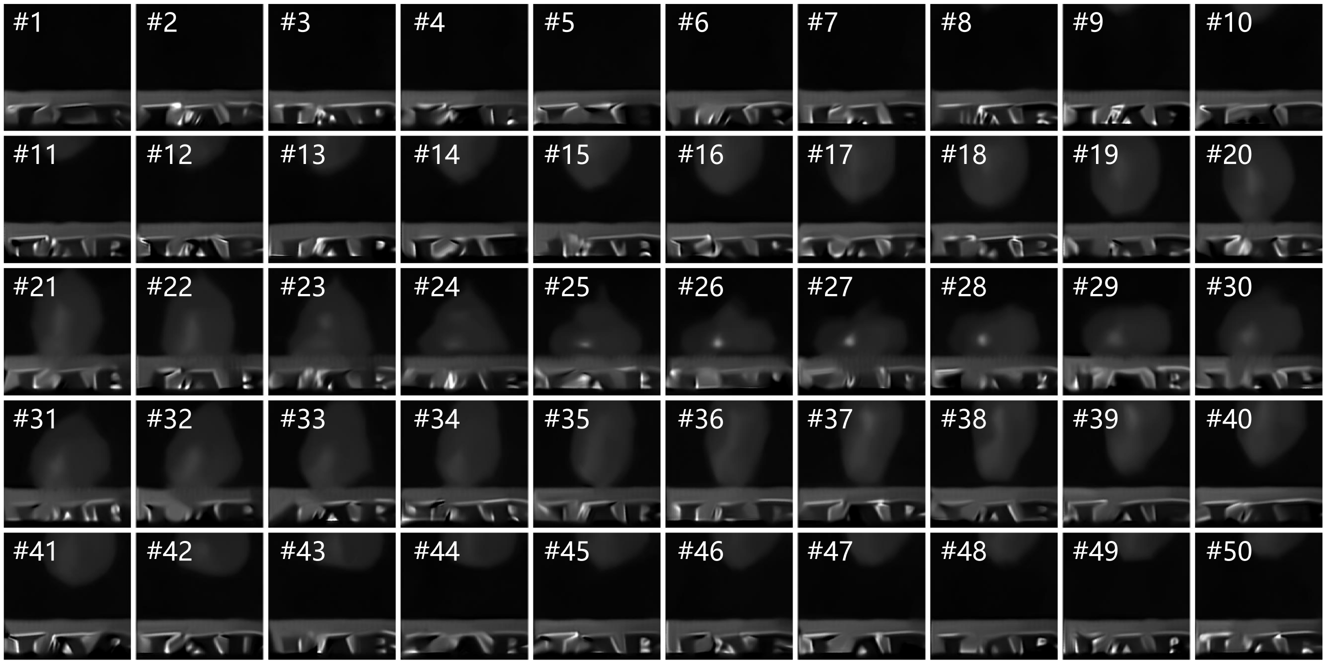

Figure 12: The results of DeSCI on real data `Water Balloon`.

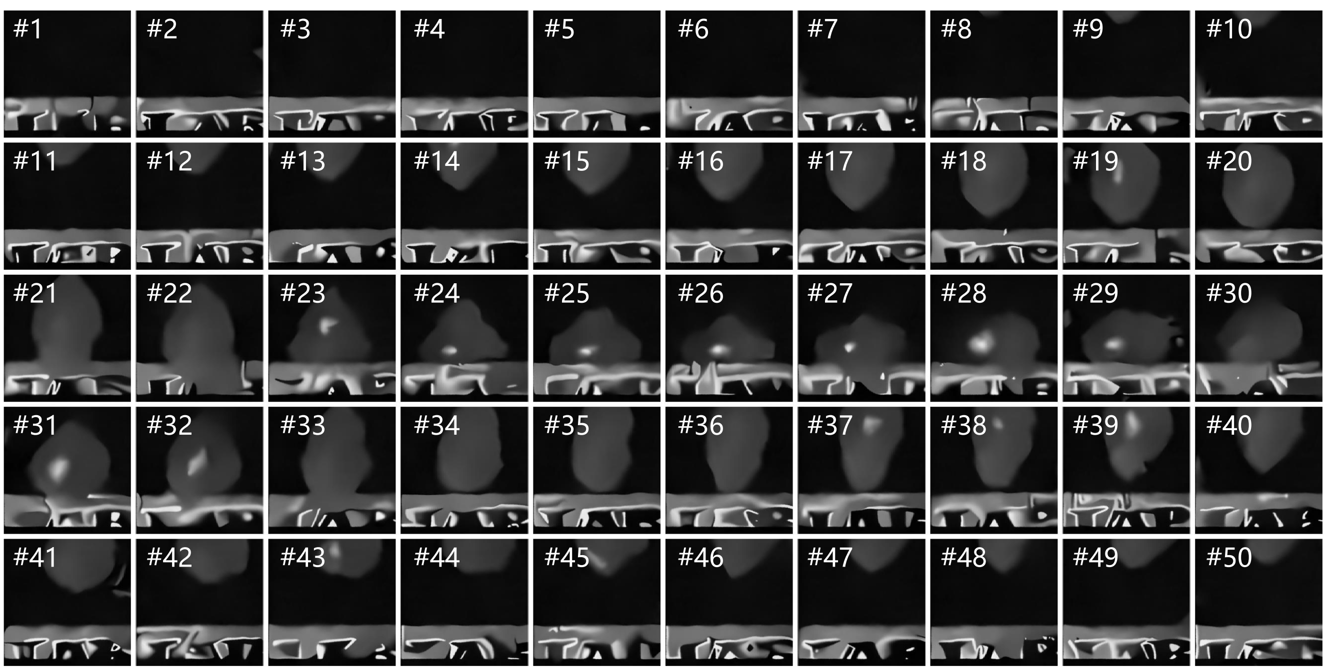

Figure 13: The results of PnP-FFDNet on real data `Water Balloon`.

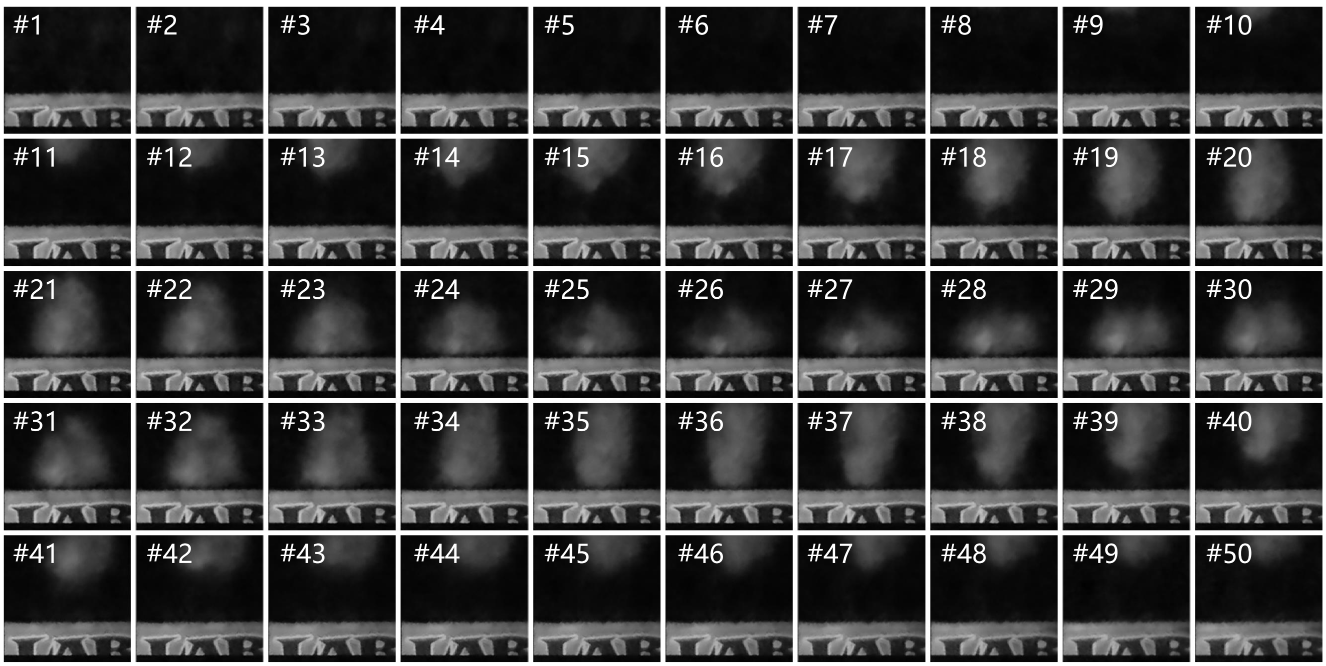

Figure 14: The results of RevSCI-net on real data `Water Balloon`.