Supplementary information for :

Towards SSD accelerating for embedded environments: a compressive sensing based approach

Imene Bouderbal*, Abdenour Amamra, M. El-Arbi Djebbar and M. Akrem Benatia.
Ecole Militaire Polytechnique, Bordj El-Bahri BP 17, Algiers, Algeria.
*Corresponding Author: imene.bouderbal@yahoo.com

This supplementary information file presents more details and experimental results, which include: S1. Full reference evaluation of the proposed lightweight compressed sensing model, S2. Qualitative results of the proposed lightweight compressed sensing model, S3. Alternative proposed approach along the obtained results, and S4. Visual detection results, respectively.

S1. Full reference evaluation of the proposed lightweight compressed sensing model :

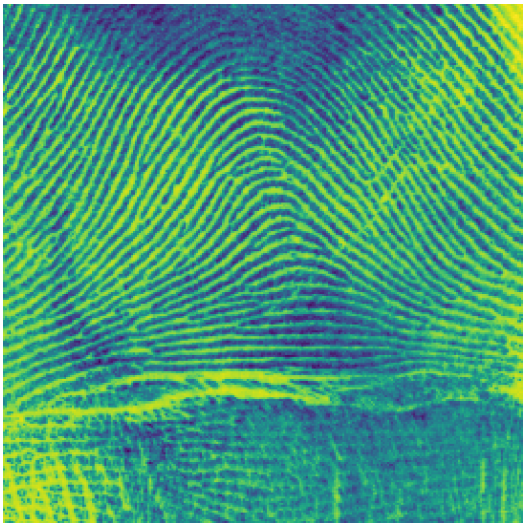
This evaluation is based on [10]. The following metrics are evaluated over the images of the SET11 dataset :

1. SSIM, the Structural Similarity index.
2. PSNR, the Peak Signal-to-Noise ratio.
3. MS-SSIM, the Multi-Scale extension of the SSIM index.
4. VIF, the Visual Information Fidelity measure.
5. FSIM, the Feature SIMilarity index.
6. GMSD, the Gradient Magnitude Similarity Deviation.
7. VSI, the Visual Saliency Induced quality index.
8. NLPD, the Normalized Laplacian Pyramid Distance.
9. LPIPS, the Learned Perceptual Image Patch Similarity model.
10. DISTS, the Deep Image Structure and Texture Similarity metric.

	SSIM	PSNR	MS-SSIM	DISTS	GMSD	LPIPS	VSI	FSIM	VIF	NLPD
best_value	1	inf	1	0	0	0	1	1	1	0
Barbara	0.6828	22.506	0.7954	0.1713	0.2318	0.4197	0.9737	0.8651	0.3534	0.3626
Boats	0.7227	23.572	0.8312	0.1657	0.1668	0.3864	0.9762	0.8831	0.3112	0.2850
Cameraman	0.7615	22.273	0.783	0.2073	0.1662	0.3971	0.9210	0.8309	0.3029	0.3026
Fingerprint	0.435	17.992	0.5228	0.2453	0.2638	0.5526	0.8870	0.7145	0.2392	0.5793
Flinstones	0.4622	14.964	0.5476	0.2535	0.2638	0.5868	0.8890	0.7004	0.1975	0.5305
Foreman	0.8431	25.302	**0.9114**	**0.1360**	0.1282	0.3484	**0.9886**	**0.9281**	0.3530	**0.1890**
House	**0.8535**	22.342	0.8761	0.1599	**0.1265**	0.3435	0.9790	0.9104	0.4009	0.1956
Lena	0.8282	**27.628**	0.8944	0.1459	0.1285	**0.3336**	0.9586	0.9129	0.3398	0.2408
Monarch	0.7484	20.809	0.7229	0.1884	0.1730	0.3686	0.9304	0.8424	0.3186	0.3139
Parrots	0.8266	20.460	0.8341	0.1815	0.1445	0.3375	0.9514	0.9117	**0.4144**	0.2543
Peppers	0.8003	23.777	0.8516	0.1704	0.1469	0.3845	0.9701	0.8942	0.3023	0.2550

Table 1: Full reference image quality metrics scores on the Set11 dataset in terms of MSE (best performance in bold, worst performance in red).

S2. Qualitative results of the proposed lightweight compressed sensing model :



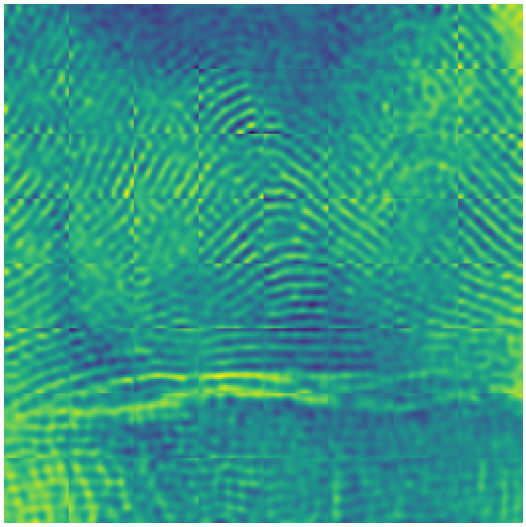


Figure 1: left : Original Set11 dataset images, right : Corresponding reconstruction using the proposed lightweight CS network trained on Div2k dataset.

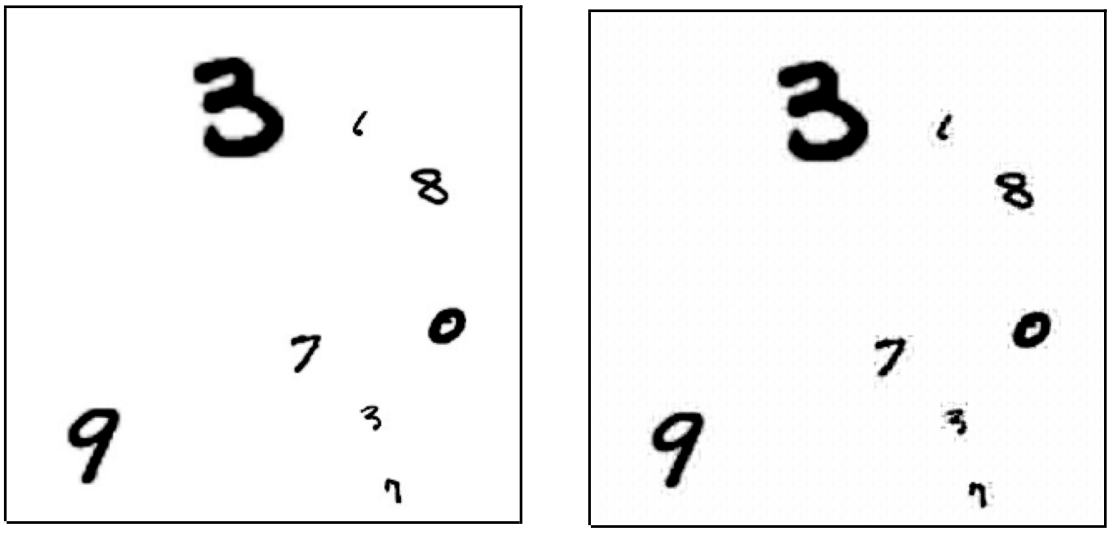


Figure 2: Sample from the YYmnist dataset : (left) original image, (right) reconstructed image.

Figure 3: Sample from the Mask dataset : (left) original image, (right) reconstructed image.

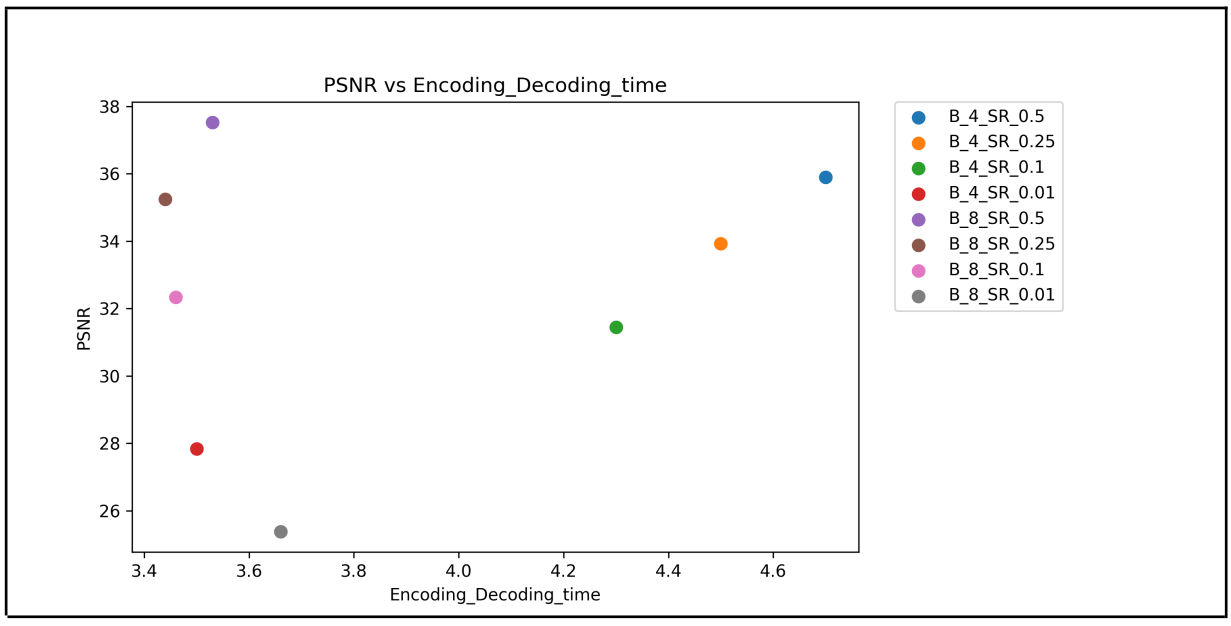


Figure 4: PSNR vs encoding and decoding time for the different trained lightweight CS networks.

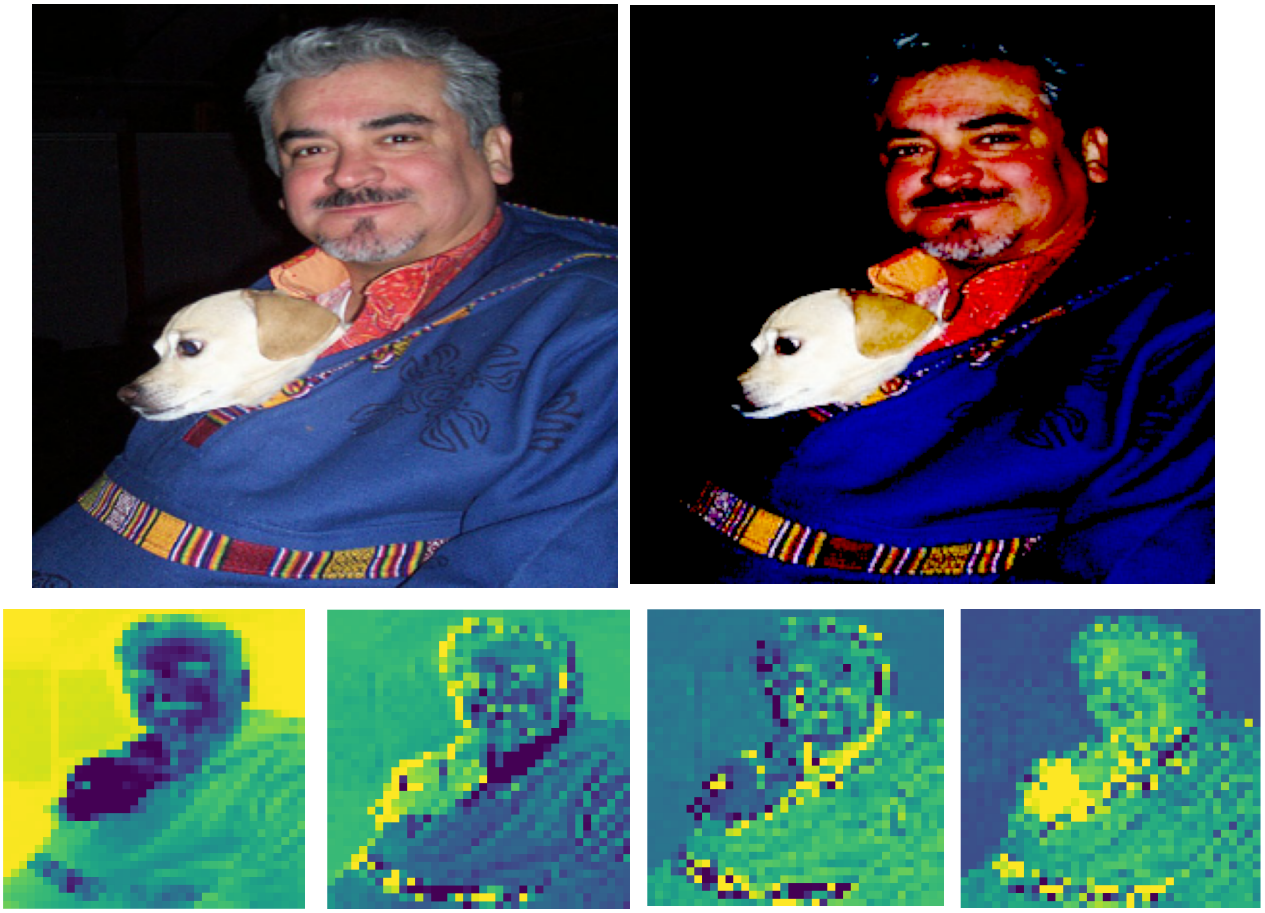


Figure 5: A sample from the Pascal VOC test set : (above left) original image, (above right) decoded image and (below) the 4 highest entropy channels of the compressed representation. These latter do not override the spatial structure of images which motivates this work.

S3. Multi-scale CS based approach

To investigate the impact of multi-scale sampling on the proposed pipeline, we propose a variant of our approach inspired by the state-of-the-art scale-space method [7] by applying minor changes to the initial procedure (refer to Figure 5 for more details). Similarly to the initial approach, the reconstruction network is linear to maintain model efficiency.

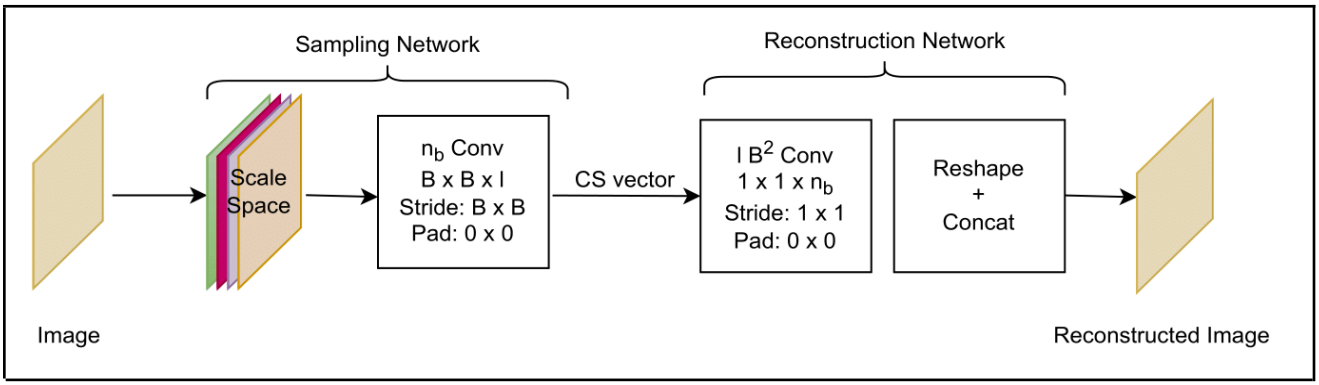


Figure 6: Proposed multi-scale CS network.

First, the image is decomposed at different scales to generate multi-scale features. The linear decomposition process, denoted scale-space, is modelled as convolution layers without activation and bias to enable end-to-end learning. we use four convolutions with the different kernels:

3×3,5×5,7×7, and 9×9 to output four decomposed features. All these layers' output is concatenated into a single output vector forming the input of the sampling stage. As for this latter, it is kept similar to the one used in the single-scale model. Table 1 summarizes the obtained results over CS_D, based on single-scale CS, and MS_D, based on multi-scale CS. For a fair comparison, both models are based on the Mobilenet backbone and use the same block size and sampling rate (B=4 and M/N=0.25).

We train both models using a large batch size of 128 for 70 epochs. We use Adam optimizer with 0.001, 0.005 and 0.0001 for the first 35 epochs, the next 15 and the last 20, respectively.

Model	**mAP**	**FPS**	**SSIM**	**PSNR**	**Enc. time**	**Dec. time**
CS_D	54.8%	81	0.989	33.932	0.0020	0.0016
MS_D	54.5%	70	0.989	33.934	0.0032	0.0019

Table 2: Results for our approach along both single-scale and multi-scale sampling.

Regarding image quality metrics, single-scale and multi-scale CS models perform equivalently. In contrast to [7], stacking a linear decomposition block did not improve the performance of the MS model when using a linear reconstruction network. In terms of accuracy and opposite to our assumption, the CS_D model performs slightly better than MS_D (54.8% vs 54.5%). Regarding Fps, the CS_D model is 13.58% faster than the MS_D one. The reason behind it is the deeper feature maps generated by the MS sampling process, which need more time to flow through the detection branch.

S4. Visual detection results

Figure 7: Some detection examples from the Pascal VOC dataset using the CS_D_8_1 model.

Figure 8 Some detection examples from the Pascal VOC dataset using the CS_D_8_25 model.

Figure 9: Some detection examples from the Mask dataset using the CS_D based VGG model.