INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2020, VOL. 66, NO. 2, PP. 347-353
Manuscript received May 7, 2019; revised April, 2020. DOI: 10.24425/ijet.2020.131884

Single Image Super-Resolution through Sparse Representation via Coupled Dictionary learning

Rutul Patel, Vishvjit Thakar, and Rutvij Joshi

***Abstract*—Single Image Super-Resolution (SISR) through sparse representation has received much attention in the past decade due to significant development in sparse coding algorithms. However, recovering high-frequency textures is a major bottleneck of existing SISR algorithms. Considering this, dictionary learning approaches are to be utilized to extract high-frequency textures which improve SISR performance significantly. In this paper, we have proposed the SISR algorithm through sparse representation which involves learning of Low Resolution (LR) and High Resolution (HR) dictionaries simultaneously from the training set. The idea of training coupled dictionaries preserves correlation between HR and LR patches to enhance the Super-resolved image. To demonstrate the effectiveness of the proposed algorithm, a visual comparison is made with popular SISR algorithms and also quantified through quality metrics. The proposed algorithm outperforms compared to existing SISR algorithms qualitatively and quantitatively as shown in experimental results. Furthermore, the performance of our algorithm is remarkable for a smaller training set which involves lesser computational complexity. Therefore, the proposed approach is proven to be superior based upon visual comparisons and quality metrics and have noticeable results at reduced computational complexity.**

***Keywords*—Single Image Super-Resolution, Dictionary Learning, Sparse representation**

I. INTRODUCTION

IMAGE Super-Resolution (SR) is an image reconstruction problem which obtains High Resolution (HR) image from given single or multiple Low Resolution (LR) images. However, in a practical scenario, multiple LR images may not be available and even if those are available, those multiple images need to be registered which is a complex process. Therefore, researchers are much focused to obtain HR image from given single LR image. Considering this, Single Image Super-Resolution (SISR) is an ill-posed problem which does not possess a unique solution due to the underdetermined system. [1]In another way, there would be many HR images which satisfy reconstruction constraint for given LR image. However, prior information about the ill-posed SR problem may mitigate the feasible solution. In a practical scenario, SR algorithms would be extremely useful to extract significant information from low-cost imaging sensors.

The SISR algorithms are primarily classified into reconstruction and learning based where reconstruction based algorithms try to interpolate the LR image in order to obtain HR image whereas learning based approach trains the dictionary and use it to obtain HR image for input test LR image. Considering learning based SISR, the coupled over-complete dictionaries (High and Low Resolution) are jointly trained from the given High and Low-Resolution training patches dataset which in turn used to reconstruct HR image. Moreover, the coupled over-complete dictionary shares the same sparse representation for the given HR-LR patch pairs.

A dictionary learning is an optimization problem involves sparse approximation and dictionary update processes which are iterated until convergence criterion satisfied. Since a decade, many algorithms for sparse approximation became popular which are Basis Pursuit (BP) [1], Matching Pursuit (MP) [2], Orthogonal Matching Pursuit (OMP) [3], Least Absolute Shrinkage and Selection Operator (LASSO) [4], Subspace Pursuit (SP) [5] and Gradient Pursuit (GP) [6]. The objective for each sparse approximation algorithm is to obtain sparse representation for a given signal through an over-complete dictionary.

An initial sparse representation is performed using an initial dictionary chosen either randomly or by simply fetching random columns of the training dataset. Through an initial dictionary, the given signal is decomposed through a linear combination of dictionary atoms i.e. dictionary columns where the weight of a dictionary atom is assigned by a sparse vector. Now in the dictionary update stage, fixing the sparse vector, the dictionary atoms are updated such that representation error is minimized. This whole process is iterated until the learned dictionary represents the training data at a satisfactory level. Previous to the dictionary learning algorithms, fixed dictionaries which has predefined mathematical transform, like Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT) and many such were used. However, due to evolving learning based approached various dictionary learning method proposed. Initially, Olshausen and Field [7] proposed a Maximum Likelihood (ML) algorithm for dictionary learning for sparse coding of natural images. However, this approach ML is further replaced by Maximum a Posteriori (MAP), proposed by Kreutz-delgado et. al. [8] which reduces computational complexity in sparse approximation stage with respect to ML [7]. Considering the same ML [7] objective function, Engan et. al. proposed a more efficient

Rutul Patel is with the Electronics and Communication engineering department as a faculty at IT, Nirma University & research scholar at Gujarat Technological University, Ahmedabad, India. (email: rutul.patel@ nirmauni.ac.in).

Vishvjit Thakar is with the department of Information & Communication Technology, Sakalchand Patel University, Visnagar, India. (email: vishvjitkthakar@gmail.com).

Rutvij Joshi is with the Electronics and Communication engineering department at A. D. Patel Institute of Technology, Vitthal Udyognagar, India. (email: ec.rutvij.joshi@adit.ac.in).

algorithm named Method of Optimized Directions (MOD) [9] which has close-form expression for dictionary update stage. Moreover, variants of MOD are also proposed such as Iterative Least Squares (ILS) [10] and Recursive Least Squares (RLS) [11]. By generalizing K-means algorithm, Aharon, Elad, and Bruckstein come up with K-means Singular Value Decomposition (K-SVD) [12] which updates single dictionary atom at a time without computing matrix inversion as required in MOD [9]. However, these approaches have a major bottleneck when the solutions are converging towards singular points rather than local minima where the objective function is not differentiable. In order to overcome this issue, Simultaneous Codeword Optimization (SIMCO) based dictionary learning algorithm is proposed by [13]. SIMCO avoids it by introducing additional regularization term in the objective function to make it differentiable. The prime objective of the SIMCO algorithm is to update sparse codes and dictionary simultaneously which enhances the learning rate.

The SIMCO based learned over-complete dictionary outperforms implementation of SISR with respect to other existing SISR algorithms [14-16] in terms of perceptual quality and quantitative metrics. Moreover, the results show that SIMCO achieves quick learning rate compared to other dictionary learning approaches. The key reason for improvement SISR results using SIMCO is that SIMCO has an additional regularization coefficient which avoids convergence of objective function at singular points.

The major contribution of this proposed algorithm as follows:

- SIMCO dictionary learning algorithm was proposed for single dictionary learning through given training set which is further applied in the image denoising problem. However, the SIMCO framework is modified into a SISR context to enable joint learning of dictionaries for given HR and LR pairs.
- Most SISR algorithms are compared with respect to quantitative metrics like PSNR and SSIM. However, the perceptual quality of an image cannot be exactly quantified through these metrics. Therefore, a quantitative metric named Weighted Signal to Noise Ratio (WSNR) [17] is used for comparison which measures the image quality based upon human visual perception.

II. DICTIONARY LEARNING

It is observed that most of the natural signals are sparsely represented exactly or approximately in any of the transform domain. The chosen dictionary to obtain transform domain representation is fixed and would not guarantee about representation error for the given set of signals. Hence, it is feasible to utilize a learning based approach where dictionary would be updated until convergence to the lowest possible representation error for a given training set of signals. To summarize, dictionary learning approach first aimed to obtain sparse representation and later on update its atoms which tries to minimize the representation error.

Consider from the training images dataset, some L patches are extracted and concatenated horizontally after converting each patch into a column vector of length N which results in training set $\boldsymbol{Y} \in R^{N\times L}$. The objective is to obtain learned overcomplete dictionary $\boldsymbol{D} \in R^{N\times K}$ which gives a sparse representation of each patch in $\boldsymbol{Y}$ with minimum possible representation error through sparse vector $\boldsymbol{X} \in R^{K\times L}$. The illustration of the dictionary learning observation model is shown in Fig. 1.

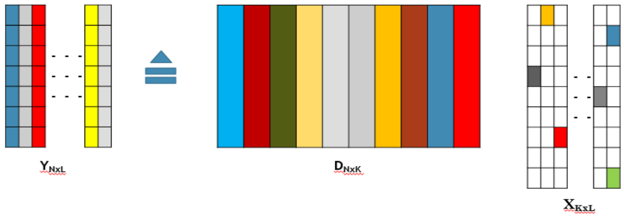

Fig. 1 Dictionary Learning: Observation Model

As illustrated in Fig.1, the dictionary $\boldsymbol{D}$ provides an approximate representation of each of the L patch in training set $\boldsymbol{Y}$ via corresponding sparse vector $\boldsymbol{X}$. The key objective is to obtain Optimized dictionary $\boldsymbol{D_{opt}}$ such that each of the L training vectors is sparsely represented as linear combination of dictionary atoms while minimizing representation error. Since each patch exhibits sparse representation, the objective function must incorporate the prior information about sparsity. Therefore, the objective function for dictionary learning can be written as,

$$\boldsymbol{D_{opt}} = \underset{\boldsymbol{D}}{argmin} \|\boldsymbol{Y} - \boldsymbol{DX}\|_2^2 + \lambda \|\boldsymbol{X}\|_1 \qquad (1)$$

The regularization coefficient λ in (1) assigns weights to a tradeoff between sparsity and representation error. The above-mentioned dictionary learning formulation (1) can be further extended for Single Image Super-Resolution problem by jointly learned HR and LR dictionary via common sparse representation which is described next.

The dictionary learning based SISR algorithm consists of training phase where one seeks for sparse representation in order to learn dictionaries (HR and LR) and later during testing phase the query LR image is super-resolved via those learned dictionaries. However, for each concatenated HR and LR patch pair there must be a common sparse vector for corresponding concatenated HR and LR dictionaries. The approach for testing hereby used is an ScSR algorithm [16] as mentioned in Algorithm 1. The ScSR algorithm is first proposed SISR algorithm which seeks for sparse representation via dictionary learning. However, due to evolving dictionary learning algorithms, the SISR results can be improved through efficiently learned dictionaries. Therefore, the SIMCO [13] based dictionary learning algorithm is imbibed into the SISR framework via jointly learned HR and LR dictionaries simultaneously.

SIMCO based dictionary learning algorithm is modified and imbibed into SISR framework to satisfy the objective of jointly learn HR and LR dictionaries. Consider a test database $\boldsymbol{Y_l}$ and $\boldsymbol{Y_h}$ created by randomly sampled LR and HR patch pairs from test images database and concatenated horizontally for each. The initial dictionary is chosen by arbitrarily choosing columns of $\boldsymbol{Y_l}$ and $\boldsymbol{Y_h}$ to obtain $\boldsymbol{D_l}$ and $\boldsymbol{D_h}$ respectively. Now, the coupled dictionary learning based on SIMCO [13] can be formulated as shown in (2) below:

$$\underset{\{\boldsymbol{D_h}, \boldsymbol{D_l}, \boldsymbol{Z}\}}{min} \frac{1}{N}\|\boldsymbol{Y_h} - \boldsymbol{D_h Z}\|_2^2 + \frac{1}{M}\|\boldsymbol{X_l} - \boldsymbol{D_l Z}\|_2^2$$
$$+\lambda\left(\frac{1}{N}+\frac{1}{M}\right)\|\boldsymbol{Z}\|_1 + \mu\left(\frac{1}{N}+\frac{1}{M}\right)\|\boldsymbol{Z}\|_2^2 \qquad (2)$$

Where, N and M are dimensions of HR and LR patch respectively in vector form, and μ is an additional regularized term to avoid singularity problem which occurs in dictionary update. Now, in order to make the expression simplified, equation (2) can be rewritten as,

$$\min_{\{\boldsymbol{D_h}, \boldsymbol{D_l}, \boldsymbol{Z}\}} \frac{1}{N}\|\boldsymbol{Y_c} - \boldsymbol{D_c}\boldsymbol{Z}\|_2^2 + \lambda\left(\frac{1}{N}+\frac{1}{M}\right)\|\boldsymbol{Z}\|_1 + \mu\left(\frac{1}{N}+\frac{1}{M}\right)\|\boldsymbol{Z}\|_2^2 \quad (3)$$

$$where, \boldsymbol{Y_c} = \begin{bmatrix} \frac{1}{\sqrt{N}}\boldsymbol{Y_h} \\ \frac{1}{\sqrt{M}}\boldsymbol{X_l} \end{bmatrix}, \boldsymbol{D_c} = \begin{bmatrix} \frac{1}{\sqrt{N}}\boldsymbol{D_h} \\ \frac{1}{\sqrt{M}}\boldsymbol{D_l} \end{bmatrix}$$

As a result of (3), we would have learned HR and LR dictionaries which is used to implement SISR as mentioned in Algorithm 1.

III. PROPOSED ALGORITHM

Consider an LR image X which can be modeled as blurred and downsampled version of HR image Y

$$\boldsymbol{X} = SH\boldsymbol{Y} \quad (4)$$

where S represents downsampling operator and H represents blurring operator.

SISR problem aims to reconstruct HR image $\boldsymbol{Y}$ from given LR image $\boldsymbol{X}$ which leads to infinite many solutions which satisfy reconstruction constraint as illustrated in (4). Therefore, sparsity prior is considered for choosing the optimum solution. In order to incorporate sparsity prior, the SISR algorithm similar to [16] based on the local and global model is used. In the local model, for each extracted LR patch, corresponding HR patch is reconstructed via sparse representation which is repeated for the entire image. Whereas, in the global model, the reconstructed LR image in the local model is updated using gradient descent algorithm to satisfy reconstruction constraint in (4). The objective of the local model is to extract high-frequency information to reconstruct the HR patch while the global model aims to reduce visual artifacts and make the image more consistent. More insight about the local and global model is described next.

A. *Local model*

For each extracted HR patch $\boldsymbol{y}$ of $\boldsymbol{Y}$, we can represent it as a sparse linear combination of learned HR dictionary $\boldsymbol{D_h}$ atoms as (5),

$$\boldsymbol{y} \approx \boldsymbol{D_h}\boldsymbol{w} \text{ for } \boldsymbol{w} \in R^K \text{ with } \|\boldsymbol{w}\|_0 \ll K \quad (5)$$

The sparse vector $\boldsymbol{w}$ will be extracted by the sparse representation of LR patch $\boldsymbol{x}$ of $\boldsymbol{X}$ through learned LR dictionary $\boldsymbol{D_l}$ by solving (6),

$$\min\|\boldsymbol{w}\|_1 \ s.t. \|F\boldsymbol{D_l}\boldsymbol{w} - F\boldsymbol{x}\|_2^2 < \varepsilon \quad (6)$$

The equivalent representation of (6) can be given as,

$$\min_{w} \|F\boldsymbol{D_l}\boldsymbol{w} - F\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{w}\|_1 \quad (7)$$

The regularization coefficient λ in (7) assigns weights to a tradeoff between sparsity and representation error. Also, linear feature extraction operator F provides perceptually meaningful constraint on sparse representation to be closest for the approximation of $\boldsymbol{x}$. As mentioned in [16], first and second order derivatives of LR patch are used as feature which are four 1D filters given as,

$$f_1 = [-1, 0, 1], \quad f_2 = f_1^T$$
$$f_3 = [1, 0, -2, 0, 1], \quad f_4 = f_3^T \quad (8)$$

These filters are applied to training images which extract edge information and encodes neighboring information.

While solving (6) for each patch, the correlation between adjacent patches is not maintained. Therefore, a one-pass algorithm as mentioned in [16] is used which is formulated as,

$$\min\|\boldsymbol{w}\|_1 \ s.t. \|F\boldsymbol{D_l}\boldsymbol{w} - F\boldsymbol{x}\|_2^2 < \varepsilon_1$$
$$and \ \|P\boldsymbol{D_h}\boldsymbol{w} - \boldsymbol{\alpha}\|_2^2 < \varepsilon_2 \quad (9)$$

Here, P extracts overlapping region between the previously reconstructed HR image and current target patch, and $\boldsymbol{\alpha}$ has values of previously reconstructed HR image with overlap. The simplified expression of (9) is given by,

$$\min_{\boldsymbol{w}} \left\|\widetilde{\boldsymbol{D}}\boldsymbol{w} - \widetilde{\boldsymbol{x}}\right\|_2^2 + \lambda\|\boldsymbol{w}\|_1 \quad (10)$$

$$\text{Where, } \widetilde{\boldsymbol{D}} = \begin{bmatrix} F\boldsymbol{D_l} \\ P\boldsymbol{D_h} \end{bmatrix} \ \& \ \widetilde{\boldsymbol{x}} = \begin{bmatrix} F\boldsymbol{x} \\ \boldsymbol{\alpha} \end{bmatrix}$$

The solution of (10) results in optimized sparse vector $\boldsymbol{w_{opt}}$ which in turn used to reconstruct HR patch for given LR patch by $\boldsymbol{y} = \boldsymbol{D_h}\boldsymbol{w_{opt}}$. It is important that dictionaries are learned to extract high-frequency textures rather than intensity levels. Hence, while acquiring a sparse representation of LR patch, mean is subtracted and added back to HR reconstructed patch. The process is iterated for each LR patch extracted in Raster-scan order and corresponding HR patch filled into HR image which in turn results into reconstructed HR image $\boldsymbol{Y_0}$.

B. *Global model*

The reconstructed HR image $\boldsymbol{Y_0}$ from the local model need not satisfy reconstruction constraint exactly due to local patch-based process. Hence, $\boldsymbol{Y_0}$ is modified to meet with reconstruction constraint (4) by projecting $\boldsymbol{Y_0}$ onto the solution space $SH\boldsymbol{Y} = \boldsymbol{X}$ as,

$$\boldsymbol{Y_{opt}} = \underset{\boldsymbol{Y}}{argmin}\|SH\boldsymbol{Y} - \boldsymbol{X}\|_2^2 + c\|\boldsymbol{X} - \boldsymbol{X_0}\|_2^2 \quad (11)$$

Using gradient descent algorithm, equation (11) can be solved by an iterative method with following update equation,

$$\boldsymbol{Y_{t+1}} = \boldsymbol{Y_t} + v[H^T S^T(\boldsymbol{X} - SH\boldsymbol{Y_t}) + c(\boldsymbol{X} - \boldsymbol{X_0})] \quad (12)$$

Here v represents the step size of gradient descent algorithm. The whole algorithm to implement SISR is described in Algorithm 1.

Algorithm 1 Coupled-dictionary learning based Single Image Super-Resolution

Input: Learned dictionaries $\boldsymbol{D_h}$, $\boldsymbol{D_l}$ and LR image $\boldsymbol{X}$.

For each extracted 5×5 patch $\boldsymbol{x}$ of $\boldsymbol{X}$ starting from the upper left corner with stride 1 scanning as raster-scan order,

- Convert the extracted patch $\boldsymbol{x}$ to be zero mean by subtracting mean $\bar{\boldsymbol{x}}$ from each pixel of the patch $\boldsymbol{x}$
- Compute sparse vector which shares same sparse representation for HR and LR patch through,
$$\underset{\boldsymbol{w}}{argmin} \|\widetilde{\boldsymbol{D}}\boldsymbol{w} - \widetilde{\boldsymbol{x}}\|_2^2 + \lambda\|\boldsymbol{w}\|_1$$
- Obtain HR patch $\boldsymbol{y} = \boldsymbol{D_h w}$
- Add mean $\bar{\boldsymbol{x}}$ into HR patch $\boldsymbol{y}$ and put in $\boldsymbol{Y_0}$

end

Through global reconstruction constraint, obtain the closest image to $\boldsymbol{Y_0}$ which satisfies,

$$\boldsymbol{Y_{opt}} = \underset{\boldsymbol{Y}}{argmin} \|SH\boldsymbol{Y} - \boldsymbol{X}\|_2^2 + c\|\boldsymbol{Y} - \boldsymbol{Y_0}\|_2^2$$

Output: SR image ***Yopt***

IV. EXPERIMENTAL RESULTS

In order to demonstrate effectiveness for the proposed algorithm, the PSNR and SSIM for standard Set14 images are computed for various SISR algorithms [14–16] (for upscale factor 2) and same is shown in TABLE I and TABLE II respectively. However, higher PSNR and SSIM values would not be always guaranteed that the reconstructed image has better perceptual quality. It is proven in the literature that a human vision system perceives certain frequency dominantly than other frequencies. Therefore, a more appropriate quantitative measure referred as Weighted Signal to Noise Ratio (WSNR) for comparison which is proposed by [18] and further modified by [17] is used for comparison. The proposed model as in [17] to compute WSNR assigns larger weights to those frequencies for which the human vision system is sensitive and lower to other frequencies. Therefore, the quality of the image is assessed based on human perceptual vision system which is justified to prove the effectiveness of the proposed algorithm. The results based on WSNR to compare various SISR algorithms for Set14 dataset are shown in TABLE III. The results show that instead of PSNR and SSIM, WSNR clearly distinguish the effectiveness of proposed SISR algorithm and it outperforms over other SISR algorithms. For training purpose, dictionary size is chosen to be 1024 which has been proven to be superior for our experiments.

For all experiments, size of the dictionary was chosen to be 1024 or 2048 to achieve a higher quality of Super-resolved image. In order to determine the most appropriate dictionary size, an experiment is performed on the set14 dataset to compute PSNR for various dictionary size as shown in Fig. 2. Additionally, the time required for learning the dictionary is also computed on a machine with Intel® Core™ i3-5005U having a 2GHz clock and 4.00GB of RAM. Considering computation time, PSNR is almost linearly increasing with respect to the size of the dictionary. The analysis shows that lower dictionary size results in poor PSNR due to corresponding sparse vector has been assigned lower dimension. Conversely, for larger dictionary size, the redundancy in the sparse vector in introduced which need to be considered while choosing the size of the dictionary. The single most striking in the result is to choose the size of the dictionary to be 1024 for best PSNR results among other dictionary sizes for the patch size of LR image to be 5 and an upscale factor of 2. Moreover, for training LR and HR dictionary, we have used a set of 91 natural images as used in [16] by randomly sampling around 25000 patches.

For overall comparision, the SISR algorithms are performed for upscale factor x2, x3 and x4 on widely used Set5 and Set14 database and quantitative parameter like PSNR, SSIM and WSNR are evaluated and their average values are mentioned in TABLE IV and TABLE V. With a few exceptions, like higher upscale factor, the proposed algorithm outperforms with respect to WNSR hence there is a scope of improvement for higher upscale factors. The key aspect for emphasizing WSNR is its direct impact on human perceptual vision and therefore the visual comparison for various set5 and set14 images are shown in Fig. 3 to Fig. 7.Observing these figures will clearly justify the use of WSNR for quantifying the effectiveness of a SISR algorithm. To produce all these experimental results, dictionary atoms are chosen to be 1024 for upscale factor 2 and 3. Since the size of overcomplete dictionary atoms is correlated with the patch size and upscale factor. Hence, in order to make dictionary overcomplete, dictionary atoms are chosen to be 2048 for upscale factor 4. In addition to that, for sparse representation, the regularization coefficient λ is selected to be 0.20 for all experiments via cross-validation and for dictionary update, the regularization parameter μ is chosen to be 0.05 as specified in [13].

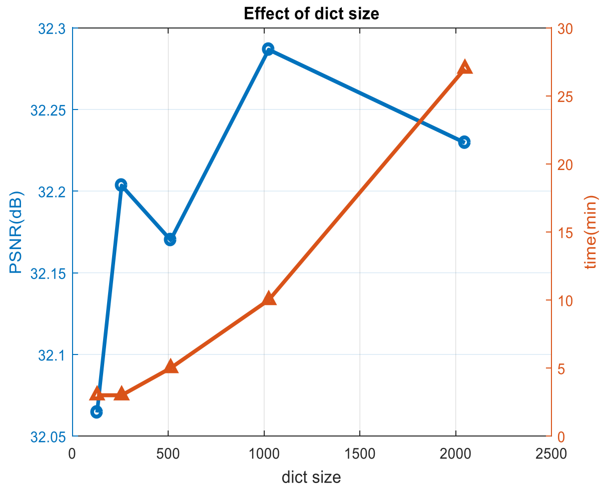

Fig. 2 Choice of dictionary size

TABLE I
PSNR results of various SISR algorithms for Set14 database (x2)

Sr No	**Image**	**nearest**	**Bicubic**	**Glasner** [14]	**SRCNN** [15]	**ScSR** [16]	**Proposed algorithm**
1	baboon	24.2037	24.6606	25.1119	**25.3626**	25.239	25.31963
2	barbara	27.1754	27.9346	28.5427	28.5021	28.527	**28.62694**
3	bridge	25.4702	26.4965	27.1901	25.8107	25.529	**27.49929**
4	coastguard	28.1945	29.1379	29.8068	30.457	30.2921	**30.5689**
5	comic	24.6056	26.0551	26.658	**28.3004**	27.6679	27.75496
6	face	33.5983	34.8348	35.2177	35.5806	35.5411	**35.61523**
7	flowers	28.4049	30.4185	31.4789	**33.0583**	32.3753	32.28662
8	foreman	30.3528	32.6673	34.1581	33.7996	**34.4633**	34.1797
9	lenna	32.3361	34.7126	35.7744	**36.4613**	36.2026	36.19169
10	man	28.0053	29.26	30.3145	**30.808**	30.4663	30.45593
11	monarch	30.1776	32.9571	36.2158	**37.1023**	35.9167	35.6
12	pepper	31.0754	33.0587	**35.0775**	33.9433	34.1208	34.16257
13	ppt3	25.0601	26.8521	29.6587	**30.2398**	28.9818	29.19893
14	zebra	27.3722	30.6785	31.1288	33.2304	32.9928	**33.30461**
Avg. PSNR		28.2880	29.9803	31.1667	**31.6183**	31.3082	31.4832

TABLE II
SSIM results of various SISR algorithms for Set14 database (x2)

Sr No	**Image**	**nearest**	**Bicubic**	**Glasner** [14]	**SRCNN** [15]	**ScSR** [16]	**Proposed algorithm**
1	baboon	0.6320	0.6368	0.6687	**0.6931**	0.6773	0.6894
2	barbara	0.8060	0.8221	0.8414	**0.8553**	0.8467	0.8530
3	bridge	0.7644	0.7922	0.8245	0.8458	0.8336	**0.8459**
4	coastguard	0.7662	0.7757	0.8087	0.8357	0.8227	**0.8388**
5	comic	0.8065	0.8436	0.8637	**0.8988**	0.8880	0.8892
6	face	0.7861	0.8011	0.8105	0.8214	0.8182	**0.8232**
7	flowers	0.8514	0.8830	0.8893	**0.8987**	0.9004	0.8966
8	foreman	0.9233	0.9427	0.9559	0.9581	**0.9589**	0.9568
9	lenna	0.8337	0.8520	0.8576	**0.8646**	0.8622	0.8636
10	man	0.8067	0.8321	0.8572	**0.8721**	0.8641	0.8678
11	monarch	0.9253	0.9509	0.9606	**0.9628**	0.9612	0.9588
12	pepper	0.8190	0.8361	0.8397	0.8402	0.8416	**0.8402**
13	ppt3	0.9172	0.9379	**0.9640**	0.9605	0.9611	0.9539
14	zebra	0.8580	0.9031	0.9114	0.9339	0.9296	**0.9351**
Avg. SSIM		0.8211	0.8435	0.8610	**0.8744**	0.8690	0.8723

TABLE III
WSNR results of various SISR algorithms for Set14 database (x2)

Sr No	**Image**	**nearest**	**Bicubic**	**Glasner** [14]	**SRCNN** [15]	**ScSR** [16]	**Proposed algorithm**
1	baboon	35.2001	35.0623	38.5720	38.0012	38.1807	**38.9902**
2	barbara	41.0867	41.6197	46.3818	44.9790	45.9852	**47.0357**
3	bridge	37.2896	37.3544	42.1315	42.4780	41.8371	**43.4151**
4	coastguard	37.3791	37.0285	40.2932	40.9303	40.3147	**41.2498**
5	comic	32.6707	32.7308	34.7162	**37.9724**	36.7538	37.6944
6	face	43.2360	43.7043	47.2435	47.4660	47.7321	**48.6852**
7	flowers	38.8596	39.5870	42.8573	45.3757	44.8940	**45.9974**
8	foreman	43.6807	44.1277	42.7750	46.1130	47.7095	**47.8486**
9	lenna	43.1062	43.8036	48.6112	49.3219	48.9729	**50.4749**
10	man	37.3833	37.5939	42.3941	**43.1214**	42.3822	**43.1214**
11	monarch	41.6187	42.8559	49.7955	49.0368	49.0810	**51.0009**
12	pepper	40.4480	40.6223	**45.9294**	43.9358	44.5011	44.7652
13	ppt3	34.1143	33.6527	38.3161	**38.5834**	37.2881	37.8051
14	zebra	39.1540	39.7838	41.5993	**46.7843**	45.4892	46.7754
Avg. WSNR		38.9448	39.2519	42.9726	43.8642	43.6515	**44.6328**

TABLE IV
Comparative analysis for various upscale (x2, x3 and x4) of Set5 dataset

Upscale	Quality metric	nearest	Bicubic	Glasner [14]	SRCNN [15]	ScSR [16]	Proposed algorithm
x2	PSNR	30.8700	33.6405	35.4073	36.2194	35.7201	35.6197
	SSIM	0.8797	0.9099	0.9243	0.9303	0.9280	0.9271
	WSNR	40.0456	40.8116	45.9146	46.4727	46.2331	**47.5869**
x3	PSNR	27.9493	30.3836	31.0747	32.3108	31.3072	31.7475
	SSIM	0.7837	0.8399	0.8512	0.8727	0.8575	0.8643
	WSNR	31.5464	32.4295	34.7974	**36.7536**	35.4485	36.6636
x4	PSNR	26.3034	28.4203	28.8167	30.0148	29.0575	29.4764
	SSIM	0.7034	0.7753	0.7832	0.8153	0.7895	0.7990
	WSNR	26.5781	27.5324	29.3800	**30.9443**	29.5205	30.4507

TABLE V
Comparative analysis for various upscale (x2, x3 and x4) of Set14 dataset

Upscale	Quality metric	nearest	Bicubic	Glasner [14]	SRCNN [15]	ScSR [16]	Proposed algorithm
x2	PSNR	28.2880	29.9803	31.1667	31.6183	31.3083	31.4832
	SSIM	0.8212	0.8436	0.8610	0.8744	0.8690	0.8724
	WSNR	38.9448	39.2519	42.9726	43.8642	43.6515	**44.6328**
x3	PSNR	25.8221	27.3102	27.9846	28.5456	27.9236	28.3063
	SSIM	0.7014	0.7421	0.7573	0.7777	0.7656	0.7738
	WSNR	31.1317	31.5984	34.081	**35.097**	34.1142	35.0657
x4	PSNR	24.4637	25.7707	26.1969	26.7702	26.153	26.4851
	SSIM	0.6176	0.6662	0.6765	0.7001	0.6850	0.6945
	WSNR	26.5312	27.0776	28.717	**29.624**	28.6844	29.3024

Fig. 3 SISR for upscale (x2) and quantitative measures PSNR,SSIM and WSNR. Left to Right: Original , Bicubic (37.05, 0.942, 46.86), Glasner (37.72, 0.946, 51.24), SRCNN (38.24, **0.952**, 51.48), SCSR (38.21, 0.950, 51.74), Proposed (**38.34**, **0.952**, **52.91**).

Fig. 4 SISR for upscale (x2) and quantitative measures PSNR,SSIM and WSNR. Left to Right: Original , Bicubic (36.68, 0.964, 39.98), Glasner (38.85, 0.967, 45.61), SRCNN (**40.28**, 0.970, 46.53), SCSR (39.70, **0.971**, 46.09), Proposed (39.55, 0.970, **47.74**).

Fig. 5 SISR for upscale (x2) and quantitative measures PSNR,SSIM and WSNR. Left to Right: Original , Bicubic (34.86, 0.801, 43.76), Glasner (35.25, 0.811, 47.15), SRCNN (35.60, 0.821, 47.49), SCSR (35.56, 0.818, 47.77), Proposed (**35.63**, **0.823**, **48.72**).

Fig. 6 SISR for upscale (x2) and quantitative measures PSNR,SSIM and WSNR. Left to Right: Original , Bicubic (34.71, 0.852, 43.80), Glasner (35.77, 0.857, 48.61), SRCNN (**36.46**, **0.864**, 49.32), SCSR (36.30, 0.862, 48.97), Proposed (36.19, **0.864**, **50.47**).

V. CONCLUSION

The proposed algorithm outperforms in terms of WSNR for upscale factor 2 on standard set5 and set14 databases compared to existing SISR algorithms. Considering PSNR and SSIM, the proposed algorithm produces better results compared to existing algorithms and comparable in the case of SRCNN. Moreover, qualitative comparison for various set5 and set14 images justifies the quality metric WSNR which is best in case of a proposed algorithm for upscale factor 2. While considering higher upscale factor like 3 and 4, the SRCNN outperforms over other algorithms and proposed algorithms are producing competitive results with respect to SRCNN. Comparing SRCNN and proposed algorithm, the SRCNN algorithm has utilized 395,909 images for training the deep neural network and hence computation cost and learning rate are significantly higher and lower respectively. Whereas, the proposed algorithm utilizes only 91 images from which merely 25,000 patches are sufficient to learn the dictionary to achieve competitive results.

In summary, we have presented coupled dictionary learning based SISR algorithm which outperforms qualitatively and quantitatively for an upscale factor of 2, while producing comparable results for higher upscale factors. We have devised SIMCO dictionary learning algorithm into SISR framework for coupled dictionary learning which outperforms with respect to SRCNN in terms of computational cost and learning rate with comparable WSNR, PSNR, and SSIM for higher upscale factors.

Results so far have been encouraging and despite this, we believe that our approach could be improved for higher upscale factors as a part of future work. In addition to this, one may explore wavelet decomposition based dictionary learning approach may yield further improvement in PSNR.

REFERENCES

[1] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic Decomposition by Basis Pursuit,” *SIAM J. Sci. Comput.*, 1998.

[2] S. G. Mallat and Z. Zhang, “Matching Pursuits With Time-Frequency Dictionaries,” *IEEE Trans. Signal Process.*, 1993.

[3] J. A. Tropp and A. C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Trans. Inf. Theory*, 2007.

[4] R. Tibshirani, “Regression shrinkage and selection via the Lasso,” *J. R. Stat. Soc. Ser. B*, 1991.

[5] W. Dai and O. Milenkovic, “Subspace pursuit for compressive sensing signal reconstruction,” *IEEE Trans. Inf. Theory*, 2009.

[6] T. Blumensath and M. E. Davies, “Gradient pursuits,” *IEEE Trans. Signal Process.*, 2008.

[7] B. A. Olshausen and D. J. Field, “Emergence of simple-cell receptive field properties by learning a sparse code for natural images,” *Nature*, 1996.

[8] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. W. Lee, and T. J. Sejnowski, “Dictionary learning algorithms for sparse representation,” *Neural Comput.*, 2003.

[9] K. Engan, S. O. Aase, and J. Hakon Husoy, “Method of optimal directions for frame design,” in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, 1999, pp. 2443–2446 vol.5.

[10] K. Engan, K. Skretting, and J. H. Husøy, “Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation,” *Digit. Signal Process. A Rev. J.*, 2007.

[11] K. Skretting and K. Engan, “Recursive Least Squares Dictionary Learning Algorithm,” *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2121–2130, 2010.

[12] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[13] W. Dai, T. Xu, and W. Wang, “Simultaneous codeword optimization (SimCO) for dictionary update and learning,” *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6340–6353, 2012.

[14] D. Glasner, S. Bagon, and M. Irani, “Super-resolution from a single image,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2009, pp. 349–356.

[15] C. Dong, C. C. Loy, K. He, and X. Tang, “Image Super-Resolution Using Deep Convolutional Networks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, 2016.

[16] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image Super-Resolution Via Sparse Representation,” *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, 2010.

[17] T. Mitsa, K. L. Varkur, and I. City, “Evaluation of Contrast Sensitivity Functions for the Formulation of Quality Measures Incorporated in Halftoning Algorithms,” *Acoust. Speech, Signal Process. 1993. ICASSP-93., 1993 IEEE Int. Conf.*, 1993.

[18] J. L. Mannos and D. J. Sakrison, “The Effects of a Visual Fidelity Criterion on the Encoding of Images,” *IEEE Trans. Inf. Theory*, 1974.