DRCS-SR: Deep Robust Compressed Sensing for Single Image Super-Resolution

HOSSAM M. KASEM\(^1\), MAHMOUD M. SELIM\(^1\), (Member, IEEE),
EHAB MAHMOUD MOHAMED\(^2,3\), (Member, IEEE), AND AMR H. HUSSEIN\(^1\)

\(^1\)Electronics and Electrical Communication Engineering Department, Faculty of Engineering, Tanta University, Tanta 31512, Egypt
\(^2\)Department of Electrical Engineering, College of Engineering, Prince Sattam Bin Abdulaziz University, Wadi Al-Dawasir 11991, Saudi Arabia
\(^3\)Department of Electrical Engineering, Faculty of Engineering, Aswan University, Aswan 81542, Egypt

Corresponding author: Ehab Mahmoud Mohamed (ehab_mahmoud@aswu.edu.eg)

ABSTRACT Compressed sensing (CS) represents an efficient framework to simultaneously acquire and compress images/signals while reducing acquisition time and memory requirements to process or transmit them. Specifically, CS is able to recover an image from a random measurements. Recently, deep neural networks (DNNs) are exploited not only to acquire and compress but also for recovering signals/images from a highly incomplete set of measurements. Super-resolution (SR) algorithms attempt to generate a single high resolution (HR) image from one or more low resolution (LR) images of the same scene. Despite the success of the existing SR networks to recover HR images with better visual quality, there are still some challenges that need to be addressed. Specifically, for many practical applications, the original images may be affected by various transformation effects including rotation, scaling, and translation. Moreover, in real-time transmissions, image compression is carried out first, followed by acquisition time reduction. To address this problem, we propose a novel robust deep CS framework that is able to mitigate the geometric transformation and recover HR images. Specifically, the proposed framework is able to perform two tasks. First, it is able to compress the transformed image with the help of an optimized generated measurement matrix. Second, the proposed framework is able not only to recover the original image from the compressed version but also to mitigate the transformation effects. The simulation results reported in this article show that the proposed framework is able to achieve high level of robustness against different geometric transformations in terms of peak signal-to-noise-ratio (PSNR) and similar structure index measurements (SSIM). For the convenience of dissemination, we make our source codes available at GitHub\(^a\).

INDEX TERMS Super-resolution (SR), compressed sensing (CS), geometric transformation, spatial transform, single image super-resolution.

I. INTRODUCTION

The traditional image acquisition system typically follows the Nyquist-Shannon sampling theorem that yields acquiring an intensive number of samples. The Nyquist-Shannon sampling theorem states that the sampling frequency should be equal or greater than twice the bandwidth of the signal. So, for efficient storage and/or transmission, compression of the signal is needed to remove redundancy by a computationally complex compression method. The data acquisition devices, in some applications, are required to be simple. Additionally, the over-sampling can damage the captured object as in medical imaging. Consequently, these kinds of image acquisition systems may not be suitable for such kind of applications.

\(^a\)https://github.com/HossamMKasem/DRCS-SR-Deep-Robust-Compressed-Sensing-for-Single-Image-Super-Resolution

The developing technology of compressed sensing (CS) presents a new technique not only for image acquisition but also for image reconstruction that can simultaneously perform the sampling and compression processes. Specifically, according to [1], [2], CS is able to recover the signal from much fewer measurements when the signal is sparse in some domain compared with the number of measurements required by Nyquist-Shannon sampling theorem. It is well known that the images can be sparsely represented as images have much amount of redundant information. In this way, according to CS theory, the images can be compressed and reconstructed efficiently. To this end, there are basic challenges that need to be solved including the design of the sampling matrix and the development of the reconstruction method.

Single image super-resolution (SISR) can be considered as one of the most famous computer vision problems that recently attracts researchers attention. SISR is an image
processing task that aims to generate a high-resolution (HR) image from its low-resolution (LR) version of images. The difficulty of achieving SISR lies in computing the HR image from its LR version as it is a many-to-one mapping problem. However, numerous image super-resolution (SR) methods have been reported to deal with this non-trivial problem [3], [4]. Over the last decade, a variety of traditional non-deep-learning (DL) based approaches have been utilized in SISR computer vision task, including prediction-based methods [5]–[7], edge-based methods [8], [9], statistical methods [10], [11], patch-based methods [12], [13], missing data reconstruction in remote sensing image [14]–[16], and sparse representation methods [10], [17].

Recently, with the massive progress achieved in the DL approaches, DL-based SR models are widely proposed and often achieve better performance compared to various traditional super-resolution methods. Specifically, convolutional neural network (CNN) methods have achieved superior performance in SISR tasks. Consequently, CNN has been presented to leverage SISR tasks [4], [18]–[35]. The correlation between the LR and HR images can be easily learned by DL-based techniques and better performance compared to conventional methods can be achieved. Additionally, these techniques are able to generate a high-resolution image. However, there are still several challenges and limitations of the existing algorithms.

A. CHALLENGES AND STATE-OF-THE-ART

In practical scenarios, our proposed framework is able to overcome the following challenges:

1) MEASUREMENT MATRIX DESIGN

The restricted isometry property (RIP) must be satisfied in the measurement matrix design in order to guarantee the performance of CS [1], [2]. Therefore, the design of the measurement matrix is crucial for spike recovery. In the last few years, various methods have been developed to design the sampling/measurement matrices. These methods are built based on random, binary, [36], [37], and structural matrices [38], [39]. However, these sampling matrices are all signal independent as well as non-optimal, due to the fact that they are unaware with the characteristics of the signal. In order to circumvent this issue, we propose a DL-based framework to design optimized sampling matrices for sensing and compressing tasks.

2) EFFICIENT RECOVERY SOLVER

In addition, CS recovery methods should be able to recover the original image with better visual quality from its compressed version. Recently, various sparsity-regularized-based methods have been presented (e.g. [40]–[42]), the greedy algorithms (e.g. [43], [44]), and the iterative thresholding algorithms (e.g. [45]). However, these algorithms are not suitable for real-time transmission due to the high computational complexity of these algorithms. Consequently, the challenge for developing recovery algorithms for real-time applications needs to be solved. In our framework, we propose to use the DL-based framework to recover the original image.

3) ROBUSTNESS AGAINST GEOMETRIC TRANSFORMATION

The images can be affected by one or many geometric transformations in many real-life applications such as translation, rotation, and scaling, etc. To establish a better relationship between the input and target output images, the spatial transformations can be utilized [46] including perspective or affine transformations to overcome the geometric transformations in all applications. Thus, the design of accurate and robust DL networks must be invariant to input spatial transformations. Despite the good performance achieved by the existing SR networks to generate the HR images, it can not alleviate the effect of the geometric transformations. This is mainly due to the limited receptive field of CNNs which makes these network unable to be spatially invariant to the position of features [47].

The spatial transformer network (STN) [46] has been presented to achieve the spatial invariance. The STN is characterized by the capability to be merged into existing CNN to provide the ability of mitigating the geometric transformation effect. Additionally, the STN network can provide a dynamic mechanism for performing a suitable transformation for each input sample [48], [49]. The spatial invariance of the SR network may not be typically non-adequately expanded by inserting the STN into the existing SR image networks. As known, the bilinear interpolation in the pixel domain is utilized by STN to estimate the transformation parameters and re-sample the input image. Thus, its output often becomes blurred compared with its original input image [46]. Consequently, we propose a robust DL-based HR image recovering framework to simultaneously address the geometric corrections and recover HR image from its corrupted version.

B. CONTRIBUTIONS AND PAPER ORGANIZATION

In summary, our main contributions are as follows:

- We propose a robust CS-SR based network that is able to simultaneously perform various tasks including; 1) compressing the input image according to the CS theory; 2) geometric transformation corrections; 3) recovering HR images.
- We propose to utilize the CS algorithm to compress the input LR images. Consequently, the processing time required to recover the HR images is reduced.
- We compare our proposed framework with existing SR image methods in terms of the training cost and learning complexity.
- The superiority of our proposed framework compared with the existing benchmarks is supported and verified in terms of both peak signal-to-noise-ratio (PSNR) and similar structure index measurements SSIM by extensive experiments.
The remainder of this article is structured as follows. Section II presents a survey on SR related work besides the basics of CS, and spatial transformations. Our proposed robust CS network is described in Section III. Section IV presents experimental results and their analysis. Finally, section V presents the conclusion of this article.

II. RELATED WORK

over the past decade, significant progress has been achieved in several applications for real-world image processing, including SISR [3], image classification [50], object detection [51], [52]. These achievements are related to the huge advances in CNNs [50], [53], computational power, and the availability of enormous amount of data [54]. This section begins by explaining related work on SISR, spatial transformations, followed by a review of CS concepts.

A. SINGLE IMAGE SUPER-RESOLUTION

Single image super-resolution (SISR) reconstruction is to recover a corresponding high-resolution (HR) image from a low-resolution (LR) image. As a result, this problem can be formulated as:

$$\hat{x}_{HR} = \arg\min_{x_{LR}} \| x_{LR} - DHx_{HR} \|_2$$

where $x_{LR}, x_{HR}$ are LR and HR images, respectively. The geometric transformations, and a down-sampling operator can be represented by degradation matrices $H$, and $D$, respectively. The down-sampling operator, $D$, is firstly applied to the HR images, (i.e., $x_{HR}$), to generate the observed LR images (i.e., $x_{LR}$). For this problem to be solved, an effective prior is needed to transform this problem into a deterministic one. In the last decades [3], [55], several solutions have been proposed to solve this ill-positioned problem. Recently a large number of DL studies have been conducted to solve the question of SR image. As a result, the existing DL-based SR techniques related to our work are evaluated as follows.

Super-resolution deep convolutional networks (SRCNN) [3] is one of the ground-breaking research which is the first attempt to apply CNN for SR problem. The main idea behind SRCNN is mapping the bicubic up-sampled LR space to HR space. Specifically, SRCNN utilizes the bicubic interpolation as a pre-processing step. Further, SRCNN uses deep convolutional layers to extract the features of the overlapping patches. Finally, the reconstructed HR images are generated by non-linearly mapping of the extracted feature vectors to each other and then aggregated in the form of patches. The SRCNN is characterized by its simple structure which utilizes only convolution layers. Thus, the input image with various sizes can be passed through the SRCNN in one path. Despite the straightforward structure of the SRCNN, it still has a number of limitations. The most critical limitation is related to the slow convergence of the network, and the network only operates on a single scale.

The authors of [56] have made great efforts to improve SRCNN’s performance through proposing FSRCNN. Specifically, the pre-processing Bicubic interpolation in SRCNN is replaced by a deconvolution post-processing activity [56]. Structurally, the FSRCNN performs extraction, shrinking, mapping, and expansion of the input by utilizing four layers of convolution. Sequentially, the FSRCNN performs the mapping step on the input followed by shrinking dimensions of the features. Finally, the expansion step is performed back at the latter point. Both SRCNN, and FSRCNN utilize the mean square errors (MSE) functions as the loss functions in network training.

Recently, the researchers tried to answer a crucial question that has appeared due to the huge utilization of CNN in SISR. The question is “Should a deeper network be implemented to optimize the SR performance?”. Among all the answers, the authors in [19] have presented an answer to this question by designing a very deep super-resolution network (VDSR). The design of VDSR is inspired by a very deep VGG-network that is used for ImageNet classification. Structurally, VDSR is a cascaded network and consists of 20 layers in depth. Furthermore, all the filters employed have a size of 3-by-3 [19]. Inspired by [50], the authors of VDSR [19] utilize the residual learning to train the VDSR network. Additionally, the weakness of SRCNN is addressed by expanding SR with a single network model to multi-scale. Despite the success of CNN based SR models, but these networks are not able to mitigate the effect of the geometric transformation. Motivated by that, we propose a deep robust network that is not only able to overcome the geometric transformation effects but also recover the HR image from its corrupted version.

The author of [21], proposed deepens the network by stacking simplified residual units on the basis of global residual, namely EDSR. This network uses global residual learning, end-to-end image super-resolution. The simplified residual unit includes only two convolution layers and one ReLU activation layer

B. SPATIAL TRANSFORMER

Invariance against geometric transformations is often a desirable property to any computer-vision model, which is also highly demanded in many practical applications across areas of both computer vision and multimedia processing. Traditionally, the researchers have proposed various methods to design a model that is invariant to affine transformations including translations, rotations, scaling, etc. Some examples of these methods that have achieved a robustness level to various transformations are hand-crafted features such as HOG [57], SIFT [58] and SCIRD [59]. Despite that the convolutional layers are able to learn the filters in a translation-invariant manner, the filter response is still not invariant. Additionally, the Max-pooling methods provide a way to be invariant to affine transformations. However, this way is not suitable for large translations. This is mainly due to the pooling in practice is performed over a small region (e.g., $2 \times 2$ or $3 \times 3$). Thus, spatial invariance is provided only up to a few pixels by each pooling. By applying filters at multiple scales and locations followed by max-pooling, a locally scale-invariant representations are obtained [60].
The spatial transformer network (STN) [46] is also closely related to our proposed work. STN is able to permit for significantly-larger (parameterized) transformations. STN can be inserted into CNN and provide an end-to-end learning mechanism. In this way, STN is able to provide a way for achieving the spatial invariance for the model. However, STN utilizes the bilinear interpolation method to sample the input images in the pixel domain. Consequently, the input images are passed through the bilinear interpolation. Thus, the quality of the transformed image deteriorates.

**FIGURE 1. Structural overview of the existing STN [46].**

Fig.1 shows a structural overview of STN [46], where three modules can be identified, including: (i) a localization network, which takes the input and estimates the transformation parameters \( \theta \); (ii) a grid generator, which creates a sampling grid for reconstruction of output images; and (iii) a sampler, which computes the output by populating the sampling grid.

**C. COMPRESSED SENSING (CS)**

The CS theory states that the vector \( \mathbf{x} \) can be accurately recovered from \( M \) random measurements over the measurement matrix [1], [2]. This can be formulated as:

\[
\mathbf{y}_{\text{CS}} = \Phi \mathbf{x}
\]

(2)

where \( \mathbf{x} \) is a signal with length \( N \) (i.e. \( \mathbf{x} \in \mathbb{R}^N \)). The measurement vector \( \mathbf{y}_{\text{CS}} \) length is equal to \( M \) and \( M << N \). Furthermore, there is a basis function \( \Psi \) in which \( \mathbf{x} \) is sparse. Consequently, the signal \( \mathbf{x} \) can be represented as \( \mathbf{x} = \Psi \hat{\mathbf{f}} \).

Then, the compressed vector can be expressed as:

\[
\mathbf{y}_{\text{CS}} = \Phi \Psi \hat{\mathbf{f}} = \mathbf{A} \hat{\mathbf{f}}.
\]

(3)

The sensing matrix, \( \mathbf{A} \), is typically full rank and should satisfy the RIP [61].

The main challenge of CS is reconstructing the original signal \( \mathbf{x} \) from the measurement vector \( \mathbf{y}_{\text{CS}} \) according to (2), which is an under-determined problem. Linear inverse problems attract a lot of attention throughout engineering and mathematical sciences. In most applications, these problems are under-determined, so one must apply additional regularizing constraints in order to obtain interesting or useful solutions. Sparsity constraints have emerged as a fundamental type of regularization [62].

The CS recovery problem can be modeled as in (4). The objective is to recover \( \hat{\mathbf{f}} \) by the knowledge of \( \mathbf{y}_{\text{CS}} \) by satisfying the following condition:

\[
\hat{\mathbf{f}} = \arg\min_{\mathbf{f}} \|\mathbf{f}\|_0 \\
\text{s.t. } \mathbf{y}_{\text{CS}} = \mathbf{A} \hat{\mathbf{f}}.
\]

(4)

As well known, this problem is NP-hard [62], [63] and needs a combinatorial search. Thus, to solve this problem, substituting the \( l_0 \)-norm by the closest convex norm, which is the \( l_1 \) norm is proposed by Chen, Donoho, and Saunders in [64]. This solution can be formulated as:

\[
\hat{\mathbf{f}} = \arg\min_{\mathbf{f}} \|\mathbf{f}\|_1 \\
\text{s.t. } \mathbf{y}_{\text{CS}} = \mathbf{A} \hat{\mathbf{f}}.
\]

(5)

During the last decades, various types of recovery algorithms were suggested including the convex optimization and greedy algorithms. The convex optimization algorithms try to convert the non-convex problem into a convex one and then an approximate solution is obtained [40]–[42], [65]. Despite the success of convex optimization to solve the CS recovery problem, it still requires a very high computational complexity. Alternatively, greedy algorithms have been proposed to reduce the computational complexity of convex optimization algorithms. The greedy algorithms, for example, include matching pursuit [43], orthogonal matching pursuit [44]. Despite the low computational complexity of the greedy algorithms compared to the convex optimization but the reconstruction quality is low. The authors of [66] proposed a non-local tensor sparse and low-rank regularization (NTSRLR) approach, which can encode essential structured sparsity of an HSI and explore its advantages for HSI-CSR task. In addition, the authors of [67] proposed the first effort to characterize the spatial and spectral knowledge using the structure-based sparsity prior. Specifically, they introduce the non-local low-rank matrix recovery model and the hyper-Laplacian prior to encode the spatial and spectral structured sparsity, respectively.

With the huge success of DL, recently, there are numerous DL-based methods that have been proposed for image CS reconstruction [68]–[72]. The authors in [71] have presented a stacked denoising autoencoder (SDA) which is able to learn the statistical dependencies between the different elements of certain signals. In this way, the signal recovery performance is improved. One of the drawbacks of the proposed SDA is that SDA suffers from the high computational complexity with increasing the signal dimension. This is mainly due to its architecture that includes a full connection between any two successive layers. By utilizing the weight sharing method, the authors in [72] presented a CNN-based reconstruction method (ReconNet). This network is able to reduce the computational complexity. Inspired by the iterative shrinkage-thresholding, the authors in [70] proposed CNN (ISTA-Net) for CS reconstruction. To this end, we can conclude that the DL-based methods run faster than the traditional image CS methods. Motivated by that, in this article, we focus on utilizing the DL-based methods to compress and recover the original images.
III. THE PROPOSED DEEP FEATURE TRANSFORMATION FRAMEWORK

The main goal of our proposed robust CS-SR network is to compress the original image and mitigate the effect of geometric transformation, simultaneously. Furthermore, our proposed framework is able to generate HR images that are similar to the original images from its compressed transformed LR images. Fig. 2 shows the flow chart of our proposed framework. Our proposed framework is able to achieve this task through the mitigation of the spatial transformer effect for distorted LR images. Specifically, the proposed framework is able to perform three tasks including (1) generating an optimized measurement matrix, (2) mitigating the geometric transformation effects, and (3) recovering HR images from its compressed version. As seen in Fig. 2, our proposed framework includes three stages starting by feeding the transformed input to the compression network which utilizes CS to compress the input. Then, the output of the compression stage is passed through the feature transformer network (FTN). The FTN is able to mitigate the geometric transformation effects by estimating the transformation parameters efficiently. Finally, the output images from the FTN are passed through an SR/CS recovery network. This network is able to generate HR images from its compressed LR version similar to the original images. To this end, an overview of our proposed framework is provided. The remaining part of this section includes more specific details of the compression network, the feature transform network, and the SR/CR recovery network.

1) COMPRESSED SENSING NETWORK

The structure of our proposed framework is shown in Fig. 3. As shown in Fig. 3, the first stage of the proposed framework is the compression network. In other words, the sampling matrix, which is used for acquiring CS measurements, is learned and optimized by the first stage. In the compression network, the image is first converted into a one-dimensional vector with size \((W * H * C) \times 1\), where \(W\), \(H\), and \(C\) are the width, height, and channel of the image. Then, a fully connected (FC) layer with \(M\) neurons is adopted to compress the image vector into a lower dimension. The weights of the FC layer are considered as the CS sampling matrix \(\Phi\). During the training step, the sampling network with the training images learns the sampling matrix. The characteristics of images can be utilized by the learned sampling matrices, and hence more image structural information can be represented in the CS measurements. Thus, the quality of the reconstructed images is increased. Furthermore, this learned sampling matrices can be employed to generate CS measurements. Finally, the compressed image vector is converted into a 3D compressed image.

2) FEATURE TRANSFORMER NETWORK

The goal of our proposed framework is to overcome the spatial transformation effects. This goal can be achieved through the mitigation of the transformation effects and then estimating the geometric parameters efficiently. To estimate the geometric parameters, we propose a feature transformer network (FTN) as shown in Fig. 3. Our proposed FTN includes three main modules including localization network, grid generator, and sampling grid. Our proposed FTN preserves the essential spatial transformer structure and the working principle. However, we have redesigned the localization network to improve the estimation of the geometric transformation parameters. In addition, our proposed FTN estimates these parameters with deep features rather than pixels.

The main contribution of our proposed framework is adding the proposed FTN as shown in Fig. 3. The FTN is considered a multi-scale deep feature mapping. In addition, to refine the features, the feature refinement unit (FRU) is added into the localization network inside the FTN. Further, we have redesigned our proposed framework to work in the feature domain rather than the pixel domain. To extract the deep features from the input image, we propose firstly passing the images through a VGG19 network. Then, to refine the extracted features, they are passed through the FRU. Structurally, FRU consists of 18 layers to provide a multi-channel feature maps. Thus, in this way, the generated multi-channel feature maps have sufficient details for estimating the optimized transformation parameters in the feature domain. Therefore, we ensure that our proposed framework performs the spatial transformations in deep features rather than pixels.
The operation of FTN as shown in Fig.3 can be illustrated in various steps. First, the input image is transformed into the feature domain from the pixel domain by passing through the VGG feature extraction network. Specifically, in our proposed FTN, we propose to use the VGG 19 [73] network to extract the deep features. By utilizing several convolution layers together with max-pooling layers, the VGG 19 is able to extract the features and increase the input image features into 512 instead of 3 features. These 512 features are utilized by FRU to estimate the transformation parameters in the feature domain. Then, by utilizing the estimated parameters, our proposed FTN interpolates the input feature domain. The operation of FTN can be formulated as provided below:

\[
F(x_{cs}) = VGG_{19}(x_{cs}, s) = F_{x_{cs}, s}
\]  

where the compressed distorted input can be represented by \(x_{cs}\), \(VGG_{19}\) represents the VGG19 network to extract deep features, and \(s\) stands for the scale of the deep feature maps. To generate an HR image similar to its ground truth and mitigate the geometric transformation effects, we design the corresponding loss function. This loss function can be illustrated as:

\[
loss = \arg \min \| DF(F_{x_{cs}, s}) - x_{HR} \|_2^2 \tag{7}
\]

where \(x_{HR}\) is the ground truth, and the output of the deep feature transformation framework can be represented by \(DF(F_{x_{cs}, s})\). Then, the localization network in Fig.3 can be driven by \(F_{x_{cs}, s}\), the extracted deep feature maps and estimating the feature transformation parameters. This step can be represented as:

\[
\hat{\theta}_F = L(F_{x_{cs}, s}) \tag{8}
\]

where \(\hat{\theta}_F\) stands for the estimated transformation parameters by using deep feature maps. Then, the localization network details with FRU are described as follows:

\[
\hat{\theta}_F = L_{FRU}(F_{x_{cs}, s}) = FC(Conv(Conv(...(F_{x_{cs}, s})))) + F_{x_{cs}, s} \tag{9}
\]

where \(F_{x_{cs}, s}\) are the input features. To estimate the feature transformation parameters \(\hat{\theta}_F\), the input features maps are refined by passing through convolutional layers to extract hierarchical levels of deep feature details. Then, the output of the last convolutional layer is added to the input according to the residual learning. The output of the summation step is fed into the final fully-connected layer, i.e., classifier, to get \(\hat{\theta}_F\).

Structurally, as shown in Fig.3, FRU consists of 18 convolutional layers. The first 3 convolutional layers are obtained from the pre-trained VGG layers and the other 15 layers are from ResNet [50] layers. The concept behind choosing some layers from VGG is ensuring a smooth transition from VGG-based deep feature extraction to feature refinement.

The estimated feature transformation parameters are fed into the transformation network \(T_F\), to obtain an estimate for the transformed feature maps \(F_{x_{cs}}\). To ensure restoration of the desired output dimension of the transformed feature map, the de-convolutional layers are added. Details of the operations are described as:

\[
F_{x_{cs}} = T_F(F_{x_{cs}}, s, \hat{\theta}_F, s) \tag{10}
\]

where \(s\) controls the deconvolutional layers.

3) SR/CR RECOVERY NETWORK

The CS theory states that if the image is well sparsely represented in a specific domain then the image can be correctly recovered from the CS measurements. So, we propose to recover the image by utilizing a CR recovery network. Our recovery network is able to achieve two tasks: 1) recovering the original image from its compressed version 2) generating HR image from its recovered version. In our proposed framework, we apply the output of our proposed FTN module to the image super-resolution module (VDSR), acting as a CS recovery network and a preprocessing unit to generate an HR image. Such a combination of FTN+VDSR not only provides a robust single image super-resolution, but also converts the multi-channel deep feature maps back into pixel domain, bridging the gap between the deep features and applications that require their output to be in pixel form.

The operation of the SR/CR recovery network can be explained as follows. First, the recovery network receives the output from the FTN, and the initial reconstructed image is obtained in the feature domain. Then, the VDSR network is used to generate HR images from its initial version. Correspondingly, our proposed deep feature transformation
framework can be described via
\[
\hat{s}_{HR} = \text{Conv}(...\text{Conv}(F_{x_i}))+F_{x_i}
\]  
(11)
where \(\hat{s}_{HR}\) represents the output transformed images in pixel domain through the feature transformation.

4) LOSS FUNCTION
The main goal of our proposed framework is to create a robust compressed sensing sensing SR network that is able to: 1) compress the original signal according to the CS theory, specifically, it is able to generate a compressed vector which is an optimized representation to the original signal; 2) alleviate the effect of spatial transforms for corrupted LR images by introducing the FTN module; 3) recovering and generating HR images from compressed LR transformed images simultaneously. Consequently, our proposed framework generates the optimized measurement matrix that is utilized for generating the optimized compressed vector \(y_{CS}\). In addition, our proposed framework not only minimizes the effect of the geometric transformation effect but also minimizes the difference between the estimated HR image and the HR image itself. Specific details are described as follows.

\[
L(\theta) = \arg\min_{\theta} \|\hat{s}_{HR} - \mathbf{x}_{HR}\|_2^2
\]
(12)
where \(L(\theta)\) is a loss function (i.e., objective function), and \(\theta\) represents the model parameters of the deep neural network. The loss function in 12 can be established via three operational steps. First, our proposed framework needs to generate the optimized measurement matrix \(A\) that is used to generate an optimized \(y_{CS}\) vector. Second, our proposed framework needs to identify the affine transformation parameters, in order to capture any possible geometric transformation, through minimizing the errors between the HR and the distorted image. Finally, we generate an estimated HR image similar to the desired one through minimizing their corresponding MSEs. As a result, our loss function can be further formulated as follows:

\[
L(\theta) = \arg\min_{\theta_A} \|y_{CS} - y_{ocs}\|_2^2 + \arg\min_{\theta_{FTN}} \|\hat{x}_{FTN} - \hat{x}_T\|_2^2 + \arg\min_{\theta_{SR/CR}} \|N(x_{cs}, \theta_{FTN}) - x_{HR}\|_2^2
\]
(13)
where \(y_{ocs}\) is the optimized compressed vector, \(\hat{x}_T\) is the output image after performing the spatial transformation, \(\hat{x}_{HR} = N(x_{cs}, \theta_{FTN})\) is the estimated HR image, \(\theta_{FTN}\) is the model parameters of the super-resolution neural network, and \(\hat{x}_T\) is the output of a deep residual learning based spatial transform module. Additionally, \(\theta_A\), \(\theta_{FTN}\), and \(\theta_{SR/CR}\) represent the compression network parameters, the estimated geometric/affine transformation parameters, and the SR/CR network parameters, respectively.

Specifically, the first part of 13 minimizes the error between the compressed vector and its optimized version. correspondingly, this part is used to obtain an optimized measurement matrix. The second part of the equation is to minimize the error between the transformed LR image and the desired HR image for the super-resolution to estimate the affine transformation parameters and mitigate the transformation effects. The last part of the equation is to minimize the error between the output of the FTN module and the HR image to obtain an image similar to the desired HR image.

IV. EVALUATIONS AND EXPERIMENTAL RESULT ANALYSIS
Extensive experiments have been carried out to evaluate the performance of the proposed framework. In addition, we report our experimental results as well as their analyses. Firstly, our proposed FTN is compared with the existing STN over the number of modeling parameters. This comparison shows that the proposed FTN overwhelms the existing STN in terms of computing cost and learning complexity. Secondly, to validate the effectiveness of our proposed framework, we have carried out a number of experiments of computer vision applications. To show the capability of our proposed framework to solve real-world problems, we apply our proposed framework to one popular computer vision task, i.e., single image super-resolution (SISR). Then, we apply our proposed framework for classification of distorted or transformed MNIST handwriting dataset to show the advantage of our proposed FTN in improving the classification performance. The following simulation results show that our proposed framework is a powerful learning tool, which is able to simultaneously handle geometric transformations of the compressed images and resolution enhancement for SISR applications.

A. EXPERIMENT SETUP
1) DATASET FOR TRAINING AND TESTING
For the training and testing of the proposed framework, we have utilized various datasets. To evaluate the proposed framework in solving the SISR computer vision problem, we have used images from the [74] and 200 images from the training set of Berkeley Segmentation dataset [75] as our training data. We augment the training data by rotation, scaling, and mirroring to increase the size of the training dataset. We have utilized the bi-cubic down-sampling and resized the images into the size of 48 \(\times\) 48. Also, we have used 5 different publicly available datasets for the testing purpose, including BSDS100 [74], SET5 [76], SET14 [77], URBAN100 [78] and MANGA109 [79]. Additionally, for the purpose of testing the performance of the proposed framework within the classification of distorted or transformed MNIST handwriting datasets, we have utilized the MNIST dataset [80] for training and testing.

2) IMPLEMENTATION DETAILS
We first simulate the effect of the geometric transformations as in [26], [81], [82] to test the proposed framework with different geometric transformation effects and under various compression ratios. Specifically, the transformed LR training images are generated by four various transformations,
including: (i) The rotation effect (R), where the original image is rotated clockwise by 20 degrees; (ii) the original image is scaled with a factor of 0.5, the scaling effect represented by S; (iii) the effect of both rotation and scaling represented by RS; (iv) translation represented by T, in which the LR images are translated by 5 pixels in both X and Y directions, and finally (v) combination effect of rotation, scaling, and translation represented by RTS. Then, the transformed images are compressed by utilizing various compression ratios including 50%, 60%, 70%, and 80%.

3) TRAINING DETAILS
During the training, our proposed framework is optimized by utilizing the stochastic gradient descent (SGD) optimization algorithm. The SGD algorithm uses a learning rate of 0.001 with no learning rate decay. We train all experiments by setting the epochs value to be 20 and the batch size is set to 25. Furthermore, our proposed framework is trained in an end-to-end manner. All the training phase is performed on NVIDIA Tesla P100. The evaluation results of all experiments are presented in terms of two widely used metrics in the SR research community, which are peak-signal-to-noise-ratio (PSNR) and Structural Similarity Index Measurements (SSIM).

4) BENCHMARKS COMPARISONS
The performance of our proposed framework is validated by comparing it with the existing state-of-the-art benchmarks. Specifically, we compare our proposed framework with existing STN [46]. Structurally, VDSR is added with our proposed framework acting as a post-processing unit, to improve the quality of the output image.

B. COMPUTATIONAL COMPLEXITY EVALUATION
We follow the design of the STN presented in [83], where STN is used to mitigate the transformation effect for the traffic line signs. Specifically, the fist convolutional layer is designed to generate 200 feature maps. Then, in the second convolutional layer, these feature maps are increased to 300. These feature maps are used to estimate the geometric transformation parameters. For our proposed framework, we utilize 16 convolutional layers to generate 64 feature maps, and two layers are used to generate 128 feature maps, and one layer to generate 512 feature maps.

Table 1 shows the comparative results between our proposed framework and the existing STN in terms of the number of model parameters. We represent the convolutional layer by Conv(k; c; n) where the variables k, n, c represent the number of feature maps, the number of filters, and filter sizes, respectively. In addition, FC (m, α) represents the fully-connected layer where the variables m, α stand for the size of the input vector, and the size of the output vector, respectively.

The results in Table 1 show that our proposed framework is more powerful in structure, more cost-effective in learning, and more capable in capturing contextual information from input images, compared to the existing STN. Moreover, it requires less number of parameters to be tuned compared to STN.

To validate the time complexity of our proposed framework, we have calculated the training time that has been taken to process the whole dataset. Additionally, we have calculated the time to train the network over one sample. Finally, we calculated the testing time for one sample and then we compare these values with the existing STN. Table 2 shows the training and testing time achieved by our proposed network and the existing STN. It is worth to mention that the calculating time is for the whole network structure not only for the FTN module. From Table 2, we can see that our proposed network may take more training time than the existing STN. This is mainly because our proposed network contains a VGG-19 network that acts as a preprocessing to convert the image from the pixel domain to the feature domain which consumes more time compared to the STN.

| TABLE 1. Comparison between the proposed FTN and the existing STN in terms of the number of parameters. |
|---|---|
| Layers | STN [46] | Proposed FTN |
| Preprocessing | Mat. Pooling(1,2) | Mat. Pooling(1,2) |
| Feature Extraction | 1x Conv(5,800,3) Rel.U | 2x Conv(3,512,64) Rel.U |
| | Mat. Pooling(2,2) | Mat. Pooling(2,2) |
| | 15200 | 589952 |
| | 15180800 | 516992 |
| | 1x Conv(3,256,64) Rel.U | 1x Conv(3,256,128) Rel.U |
| | 73856 | 73792 |
| Fully Connected layer | PC(32768,1024) | FC(1024,30) |
| Classifier | FC(200,4) | FC(30,6) |
| Total parameters | 34278000 | 9544068 |

| TABLE 2. Experimental training and testing times achieved by STN compared with our proposed framework. |
| Network/Time | Train time/complete dataset | Total Train Time/sample | Testing time/sample |
|---|---|---|
| | | | |
| STN | 45 hours | 63.85 ms | 26.42 ms |
| Ours | 50 hours | 75 ms | 50 ms |

Since the processing time is not only affected by the number of parameters but also dependent on the floating-point operations (FLOPs), we have compared the FLOPs number of our proposed framework with the existing STN. We have followed [84] for calculating the number of FLOPs. As well known, the convolutional and FC layers are the most computationally expensive parts in the network. These layers perform huge numbers of multiplication and addition processes. Consequently, most of the FLOPs are consumed in these layers. Specifically, the computational cost for the convolution operation is \( k \times k \times n \times c \times W \times H \) where \( k, n, c, \) and \( W \times H \) denote the filter size, the number of the input feature channels, the number of the convolutional filters, and the size of the input feature. Additionally, the computational cost of the FC layer is \( N_{in} \times N_{out} \), where \( N_{in} \) and \( N_{out} \) are the numbers of the input and output neurons, respec-
tively. Therefore, the number of FLOPs of the convolutional layers depends on the size of the input images. We have selected one of the compression ratios (e.g., CR = 50%), and then we calculate the number of FLOPs of our proposed framework and the existing STN. Specifically, our proposed frameworks require 616.21 million FLOPs at CR = 50%. On the other side, the existing STN requires 4209.95 million FLOPs at the same CR. Therefore, it is clear that STN requires a higher number of FLOPs as its convolutional layer generates a huge number of feature maps. As noticed from Table 1, we can see that the STN localization network includes three convolutional layers that generate 200, 300, 200 features maps, respectively. As mentioned in Section IV-A3 that we are training our model on a powerful GPU, namely Tesla P 100 for PCIe, it is able to perform a few teraFlops per second. Therefore, our training process can be performed easily. In general, fortunately, the neural networks can be first off-line trained using powerful GPUs and then online fine-tuned, which can be accelerated by techniques, e.g., meta-learning.

C. EXPERIMENTS ON EFFECTIVENESS AND ROBUSTNESS OF OUR PROPOSED NETWORK

To validate the performance of our proposed framework, we have conducted extensive experiments. First, we compare the performance of our proposed framework with the existing SR state-of-the-art networks including SRCNN [3], FSRCNN [56], VDSR [19], EDSR [21]. Specifically, we have compressed the original images then we have utilized these networks to recover the original images.

Fig.4 shows ten natural images that are used to test the performance of our proposed framework. These images are selected from various images in the test datasets (i.e., Set5 [76], Set14 [77]). First, the original images have been transformed utilizing the transformation effects as mentioned in IV-A2. Then, the robustness of our proposed framework against geometric transformations is explored by comparing it with various SR state-of-the-art networks.

First, the original images are rotated by 20 degrees. Then, in our comparison, we set the compression ratio value to be 50%. Fig. 5 shows the objective evaluation in terms of PSNR and SSIM values where the original image is rotated by 20 degrees with utilizing a 50% compression ratio. From Fig. 5, we can observe the superiority of our proposed framework compared with the state-of-art networks.

To validate the performance of our proposed framework with a strong transformation effect. We have transformed the original image using a combination of rotation, translation, and scaling effects. The PSNR and SSIM values are shown in Fig.6. From Fig.6, it can be confirmed that our proposed framework is able to perform the compression and decompression of the transformed images efficiently compared to the existing state-of-the-art.

We have conducted extensive experiments to test the performance of our proposed framework compared with the existing state-of-the-art with the various testing datasets. Specifically, we have transformed the original images using two versions of the transformation effects including the rotation effect, and the combination of the rotation, translation, and scaling. Then, we test the performance of our proposed framework with the testing datasets as in IV-A1. Table 3 shows the experimental results in terms of PSNR and SSIM.
From Table 6, we can conclude that our proposed framework has a higher PSNR and SSIM values. This is mainly because our proposed framework contains a spatial module that is able to overcome the effect of the geometric transformation. Additionally, our framework includes the SR/CR recovery network that is able to recover the original images from its compressed transformed images. On the other side, the existing state-of-the-art network is able to generate an HR image from compressed images without the ability to mitigate the effect of geometric transformation. Consequently, the performance of the existing SR networks is degraded with these transformations.
In the next section, we propose to use one existing module which is utilized to overcome the geometric transformation and then compare its performance with our proposed framework.

2) EXPERIMENTS ON EFFECTIVENESS AND ROBUSTNESS OF OUR PROPOSED NETWORK COMPARED WITH EXISTING STN

To test the effectiveness and accuracy of our proposed framework, we have carried out a range of experiments using...
natural images. All the experiments show the ability of our proposed framework to estimate the affine transformation parameters and to overcome the geometric transformation effects from the compressed version of LR images. Fig.4 shows 10 natural images that are used to test the performance of our proposed framework. These images are selected from various images in the test datasets (i.e., Set5 [76], Set14 [77]).

The performance of our proposed framework is evaluated by comparing it with benchmarks that are presented in IV-A4. First, the original images have been transformed utilizing the transformation effects as mentioned in IV-A2. Then, the robustness of our proposed framework against geometric transformations is explored by comparing it with the STN [46].

The objective evaluation results in terms of PSNR/SSIM are shown in Fig. 7, 8, 9 where the original images are transformed using various transformation effects. Specifically, the original images are rotated by 20 degrees. Then, we compress the rotated images by a ratio of 50%. From Fig.7, we can conclude that our proposed framework is able to achieve better PSNR/SSIM values compared to the existing STN benchmarks.

For more convenience, we tested the performance of our proposed framework with different transformation effects as we have transformed the original images by translation effect. Specifically, the original images are rotated by 20 degrees. Then, we compress the rotated images by a ratio of 50%. From Fig.7, we can conclude that our proposed framework is able to achieve better PSNR/SSIM values compared to the existing STN benchmarks.

For more convenience, we tested the performance of our proposed framework with different transformation effects as we have transformed the original images by translation effect. Specifically, the original images are rotated by 20 degrees. Then, we compress the rotated images by a ratio of 50%. From Fig.7, we can conclude that our proposed framework is able to achieve better PSNR/SSIM values compared to the existing STN benchmarks.

The robustness of our proposed framework is tested against various transformation effects and different compression ratios. We transformed the original images with simulated geometric transformations that are mentioned previously in Sec.IV-A2, including rotation (R), scaling (S), rotation and scaling (RS), translation (T), and combination of rotation, scaling, and translation (RTS). In addition, we have compressed the original images using various compression ratios, e.g., 50%, 60%, 70%, and 80%. To this end, we have evaluated the performance of our proposed framework in five experiments compared to the existing state-of-the-art STN networks [46]. Then, we show the ability of our proposed framework to recover HR images from its compressed LR images.

Structurally, we add the VDSR with the existing STN to formulate a new benchmark, referred to as STN-VDSR. Tables 4, 5, 6, 7, and 8 show experimental results of PSNR/SSIM and SSIM values under the Rotation and scaling (RS) effects.

The robustness of our proposed framework is tested against various transformation effects and different compression ratios. We transformed the original images with simulated geometric transformations that are mentioned previously in Sec.IV-A2, including rotation (R), scaling (S), rotation and scaling (RS), translation (T), and combination of rotation, scaling, and translation (RTS). In addition, we have compressed the original images using various compression ratios, e.g., 50%, 60%, 70%, and 80%. To this end, we have evaluated the performance of our proposed framework in five experiments compared to the existing state-of-the-art STN networks [46]. Then, we show the ability of our proposed framework to recover HR images from its compressed LR images.

Structurally, we add the VDSR with the existing STN to formulate a new benchmark, referred to as STN-VDSR. Tables 4, 5, 6, 7, and 8 show experimental results of PSNR/SSIM and SSIM values under the Rotation and scaling (RS) effects.

The robustness of our proposed framework is tested against various transformation effects and different compression ratios. We transformed the original images with simulated geometric transformations that are mentioned previously in Sec.IV-A2, including rotation (R), scaling (S), rotation and scaling (RS), translation (T), and combination of rotation, scaling, and translation (RTS). In addition, we have compressed the original images using various compression ratios, e.g., 50%, 60%, 70%, and 80%. To this end, we have evaluated the performance of our proposed framework in five experiments compared to the existing state-of-the-art STN networks [46]. Then, we show the ability of our proposed framework to recover HR images from its compressed LR images.

Structurally, we add the VDSR with the existing STN to formulate a new benchmark, referred to as STN-VDSR. Tables 4, 5, 6, 7, and 8 show experimental results of PSNR/SSIM and SSIM values under the Rotation and scaling (RS) effects.
TABLE 7. The experimental results achieved by the existing STN compared with our proposed framework in terms of PSNR/SSIM values under the translation (T) effect.

| Dataset/CR | Our Proposed | Existing STN+VDSR [46] | Our Proposed | Existing STN+VDSR [46] | Our Proposed | Existing STN+VDSR [46] | Our Proposed | Existing STN+VDSR [46] |
|------------|--------------|-------------------------|--------------|-------------------------|--------------|-------------------------|--------------|-------------------------|
| BSD        | 29.90/0.8792 | 19.46/0.7248            | 25.20/0.6742 | 17.57/0.7018            | 24.44/0.8412 | 14.46/0.6883            | 23.28/0.4840 | 13.85/0.6080            |
| Lena       | 25.02/0.9037 | 19.74/0.7524            | 23.10/0.8891 | 17.06/0.7992            | 22.66/0.8582 | 11.95/0.8742            | 20.92/0.3117 | 11.60/0.5680            |
| Set14      | 22.87/0.9310 | 16.42/0.7466            | 21.05/0.9221 | 16.74/0.7385            | 20.55/0.9155 | 11.07/0.7107            | 19.70/0.8294 | 13.42/0.6292            |
| Urban      | 22.46/0.9023 | 19.26/0.7413            | 25.44/0.8922 | 16.52/0.7118            | 24.61/0.8866 | 12.96/0.6890            | 23.13/0.8687 | 12.55/0.6157            |

all CRs. Accordingly, it can be concluded that our proposal successfully provides a well-validated solution for tackling the effects of geometric transformations, and achieves a robust single image super-resolution for compressed sensing networks.

We further illustrate a number of samples in Fig.12, and 13 to compare our proposed framework with the existing STN for visual inspections and subjective assessments. Specifically, in Fig.12, we show the results on BSDS 100, URBAN 100, SET 5, and SET 14 with CR = 50% under the rotation(R) effect. From these results, we can see that our proposed framework accurately mitigates the rotation effect, and recover HR images efficiently compared to the existing STN. In addition, our proposed framework is able to accurately recover straight lines and the grid patterns well, such as the stripes on the tiger.

We test our proposed framework with combinational effects of rotation, scaling and translation (RTS) to show the robustness of our proposed FTN under more than one transformation effect. Moreover, Fig.13 shows the visual comparisons under combinational effects of rotation, scaling and translation (RTS) with CR = 50%. From Fig.13, our proposed framework is able to reconstruct the details of the numbers, eye of the baby and the word precisely. On the contrary, STN+VDSR fails to reconstruct the numbers, eye of the baby and the words clearly.

3) EXPERIMENT ON CONVERGENCE

We also test the convergence of the proposed framework according to the loss function (7) compared to the STN-VDSR and the convergence curves during the training, and testing steps are presented. In addition, the performance of our proposed framework is tested by calculating the PSNR/SSIM values during each epoch to compare these values with the existing STN.

The convergence of our proposed framework under the rotation(R) effect and CR = 50%, and the corresponding PSNR/SSIM values are presented in Fig. 10. The results in Fig.10 indicate that our proposed framework has lower loss function values compared to the existing STN.
Furthermore, the PSNR/SSIM curves validate the superiority of our proposed VDST in terms of both PSNR and SSIM values. To provide a comprehensive assessment, the convergence of our proposed framework under a stronger transformation effect which is a combination of rotation, translation and scaling (RTS) with CR = 50% is tested, and it is compared with the existing STN as shown in Fig.11. From the results shown in Fig.11, it can be confirmed that our proposed framework has lower loss function values and higher PSNR/SSIM compared with the existing STN.

4) EXPERIMENT ON IMAGE CLASSIFICATION
To show the power of our proposed framework to facilitate robust image classification against the geometric...
TABLE 9. Experimental classifications accuracy results in image classification (percentage).

| Transformation effect | Rotation (R) effect | Translation (T) effect | RTS effect |
|-----------------------|--------------------|-----------------------|-----------|
|                       | Existing STN  | Our proposed    | Existing STN | Our proposed | Existing STN | Our proposed | Existing STN  | Our proposed    |
| 50                    | 98.10        | 99.11            | 98          | 99.62        | 98.26        | 98.56        | 97.97        | 99.00            | 97.58        | 99.32        |
| 60                    | 97.00        | 99.10            | 97.28       | 98.58        | 97.54        | 98.64        | 96.69        | 98.30            | 96.83        | 98.50        |
| 70                    | 96.54        | 98.4             | 96.64       | 98.26        | 96.50        | 98.57        | 96.17        | 97.75            | 96.62        | 98.11        |
| 80                    | 95.32        | 97.99            | 95.78       | 97.55        | 95.98        | 98.04        | 95.97        | 97.11            | 95.41        | 97.69        |

FIGURE 14. Samples from MNIST dataset used for the classification purpose. (a) The original images (b) The distorted images by rotation, translation and scaling effects.

V. CONCLUSION

In this article, we have presented a compressed-sensing based framework that shows higher robustness against the geometric transformation effects. Our proposed framework has shown immunity against the weaknesses of the widely-adopted STN. Our proposed framework owns a number of features over the existing STN, including 1) our proposed framework is able to extract the content-characterized deep features out of the compressed input LR in feature domain to estimate the geometric transformation parameters and control the super-resolution process; 2) our proposed FRU is able to refine the multi-channel feature maps such that more details and their hierarchies can be revealed and provided to achieve better construction of the output images as well as a higher level of robustness; 3) our proposed framework is able to enhance the construction of the output images with higher level of robustness. Extensive experiments and comparative analysis have proven the powerful capability of the proposed DFTN to tackle complex computer vision problems in comparison with the existing STN, such as robust single image super-resolution, and image classification.

REFERENCES

[1] D. L. Donoho, “Compressed sensing,” IEEE Trans. Inf. Theory, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[2] E. J. Candes and T. Tao, “Near-optimal signal recovery from random projections: Universal encoding strategies?” IEEE Trans. Inf. Theory, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
[3] 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, Feb. 2016.
[4] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, “Image super-resolution using very deep residual channel attention networks,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 286–301.
[5] R. Keys, “Cubic convolution interpolation for digital image processing,” IEEE Trans. Acoust., Speech, Signal Process., vol. 29, no. 6, pp. 1153–1160, Dec. 1981.
[6] C. E. Duchon, “Lanczos filtering in one and two dimensions,” J. Appl. Meteorol., vol. 18, no. 8, pp. 1016–1022, Aug. 1979.
[7] M. Irani and S. Peleg, “Improving resolution by image registration,” CVGIP, Graph. Models Image Process., vol. 53, no. 3, pp. 231–239, May 1991.
[8] G. Freedman and R. Fattal, “Image and video upsampling from local self-examples,” ACM Trans. Graph., vol. 30, no. 2, p. 12, 2011.
[9] J. Sun, Z. Xu, and H.-Y. Shum, “Image super-resolution using gradient profile prior,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2008, pp. 1–8.
[10] Z. Xiong, X. Sun, and F. Wu, “Robust Web image/video super-resolution,” IEEE Trans. Image Process., vol. 19, no. 8, pp. 2017–2028, Aug. 2010.
[11] W. T. Freeman, T. R. Jones, and E. C. Pasztor, “Example-based super-resolution,” IEEE Comput. Graph. Appl., vol. 2, no. 2, pp. 56–65, Aug. 2002.
[12] N. Chang, D.-Y. Yeung, and Y. Xiong, “Super-resolution through neighbor embedding,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), vol. 1, Jul. 2004, p. I.
[13] D. Glasner, S. Bagon, and M. Irani, “Super-resolution from a single image,” in Proc. IEEE 12th Int. Conf. Comput. Vis., Sep. 2009, pp. 349–356.
[14] Q. Zhang, Q. Yuan, C. Zeng, X. Li, and Y. Wei, “Missing data reconstruction in remote sensing image with a unified Spatial–Temporal–Spectral deep convolutional neural network,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 8, pp. 4274–4288, Aug. 2018.
[15] Q. Zhang, Q. Yuan, J. Li, X. Liu, H. Shen, and L. Zhang, “Hybrid noise removal in hyperspectral imagery with a Spatial–Spectral gradient network,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 10, pp. 7317–7329, Oct. 2019.
[16] Q. Zhang, Q. Yuan, J. Li, Z. Li, H. Shen, and L. Zhang, “Thick cloud and cloud shadow removal in multitemporal imagery using progressively spatio-temporal patch group deep learning,” ISPRS J. Photogramm. Remote Sens., vol. 162, pp. 148–160, Apr. 2020.
[17] J. Yang, J. Wright, T. Huang, and Y. Ma, “Image super-resolution as sparse representation of raw image patches,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2008, pp. 1–8.
W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, “Deep Laplacian

B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, “Enhanced deep residual

J. Kim, J. K. Lee, and K. M. Lee, “Deeply-recursive convolutional network

J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using

C. Dong, C. L. Chen, K. He, and X. Tang, “Learning a deep convolutional

H. M. Kasem

S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, “Sparse reconstruction

I. Daubechies, M. Defrise, and C. De Mol, “An iterative thresholding

A. Amini and F. Marvasti, “Deterministic construction of binary, bipo-

J. S. Park, J. W. Soh, and N. I. Cho, “High dynamic range and

Y. Zhao, G. Li, W. Xie, W. Jia, H. Min, and X. Liu, “GUN: Gradual

P. Liu, Y. Hong, and Y. Liu, “Deep differential convolutional network for

F. Li, H. Bai, L. Zhao, and Y. Zhao, “Dual-streams edge driven encoder-

Y. Wang, L. Wang, H. Wang, and P. Li, “End-to-End image super-

Z. Lu, Z. Yu, P. Yali, L. Shigang, W. Xiaojun, L. Gang, and R. Yuan, “Fast

Z. Liu, Z. Yu, P. Yali, L. Shigang, W. Xiaojun, L. Gang, and R. Yuan, “Fast

Y. Wang, L. Wang, H. Wang, and P. Li, “End-to-End image super-

Y. Zhao, W. Li, Y. Zhang, and Z. Feng, “Residual super-resolution single

L. Zhao, Q. Sun, and Z. Zhang, “Single image super-resolution based on

W. Lu, T. Dai, and S.-T. Xia, “Binary matrices for compressed sensing,” IEEE Trans. Signal Process., vol. 66, no. 1, pp. 77–85, Jan. 2018.

A. Amini and F. Marvasti, “Deterministic construction of binary, bipol-

X. Gao, J. Zhang, W. Che, X. Fan, and D. Zhao, “Block-based compressive

K. Q. Dinh, H. J. Shim, and B. Jeon, “Measurement coding for compres-

I. Daubechies, M. Defrise, and C. De Mol, “An iterative thresholding

S. M. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by

J. Skilling and S. Gull, “Algorithms and applications,” in Maximum-

K. Q. Dinh, H. J. Shim, and B. Jeon, “Measurement coding for compres-

Q. Zhang, J. Zhang, and S. G. Mallat, “Matching pursuits with time-frequency

J. A. Tropp and A. C. Gilbert, “Signal recovery from random measure-

J. Haupt and R. Nowak, “Signal reconstruction from noisy random projec-

M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, “Spatial

K. Lenc and A. Vedaldi, “Understanding image representations by mea-

J. Huang and K. Murphy, “Efficient inference in occlusion-aware generative

D. Jimenez Rendeze, S. Mohamed, I. Danihelka, K. Gregor, and D. Wierstra, “One-shot generalization in deep generative models,” 2016, arXiv:1603.05106. [Online]. Available: http://arxiv.org/abs/1603.05106

K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 770–778.

R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis., Oct. 2015, pp. 1440–1448.

K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis., pp. 2961–2969, 2017.

G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jul. 2017, pp. 4700–4708.

I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard, “The MegaFace benchmark: 1 million faces for recognition at scale,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 4873–4882.

J. Jiang, X. Ma, C. Chen, T. Lu, Z. Wang, and J. Ma, “Single image super-resolution via localized regularized anchored neighborhood regression and nonlocal means,” IEEE Trans. Multimedia, vol. 19, no. 1, pp. 15–26, Jan. 2017.

C. Dong, C. L. Chen, and X. Tang, “Accelerating the super-resolution

N. Dalal and B. Triggs, “Histograms of oriented gradients for human detec-

E. J. Candés, “The restricted isometry property and its implications for compressed sensing,” Comp. Rendus Mathematique, vol. 346, nos. 9–10, pp. 589–592, May 2008.

E. J. Candés and T. Tao, “Decoding by linear programming,” IEEE Trans. Inf. Theory, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

J. Skilling and S. Gull, “Algorithms and applications,” in Maximum-

S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by

J. Xue, Y. Zhao, W. Liao, and J. Chan, “Nonlocal tensor sparse repre-

J. Xue, Y. Zhao, W. Liao, and J. Chan, “Hyper-laplacian regularized nonlocal low-rank matrix recovery for hyperspectral image compressive sensing reconstruction,” Remote Sens., vol. 11, no. 2, p. 193, Jan. 2019.

J. Xue, Y. Zhao, W. Liao, and J. C.-W. Chan, “Hyper-laplacian regularized nonlocal low-rank matrix recovery for hyperspectral image compressive sensing reconstruction,” Int. J. Imag. Syst. Technol., vol. 31, no. 4, pp. 406–420, Oct. 2019.
HASSAN M. KASEM received the B.Sc. degree (Hons.) in electronics and electrical communications engineering from the Faculty of Engineering, Tanta University, Egypt, and the M.Sc. and Ph.D. degrees in electronics and communication engineering from the Egypt Japan University of Science and Technology, in 2012 and 2015, respectively. From 2014 to 2015, he was a Researcher with the Department of Advanced Information Technology, Graduate School of ISEE, Kyushu University, Japan. Since 2015, he has been serving as an Assistant Professor with the Department of Electronics Engineering and Electrical Communications, Faculty of Engineering, Tanta University. He is currently a Postdoctoral Research Fellow with the College of Information Engineering, Shenzhen University, China. His research interests include radio resource allocation for 5G heterogeneous networks, 5G new multi-carrier modulation waveforms, massive MIMO precoding and detection, and nonorthogonal multiple access and its applications of signal processing in wireless communications.

EHAB MAHMOUD MOHAMED (Member, IEEE) received the B.E. degree in electrical engineering and the M.E. degree in computer science from South Valley University, Egypt, in 2001 and 2006, respectively, and the Ph.D. degree in information science and electrical engineering from Kyushu University, Japan, in 2012. From 2013 to 2016, he was with Osaka University, Japan, as a Specially Appointed Researcher. Since 2017, he has been an Associate Professor with Aswan University, Egypt. He is currently on leave from Aswan University and works as an Associate Professor with Prince Sattam Bin Abdulaziz University, Saudi Arabia. His current research interests include 5G networks, cognitive radio networks, millimeter wave transmissions, Li-Fi technology, MIMO systems, and underwater communication. He is also a technical committee member of many international conferences and a reviewer of many international conferences, journals, and transactions. He is also the General Chair of the IEEE ITEMS 2016 and the IEEE ISWC 2018.

AMR H. HUSSEIN received the B.Sc. degree in electronics and electrical communications engineering and the M.Sc. and Ph.D. degrees from Tanta University, Egypt, in 2001, 2007, and 2012, respectively. His M.Sc. degree was dedicated to the VHDL implementation of both multiple access interference (MAI) and inter-symbol interference (ISI) cancellers for DS-CDMA communications using field-programmable gate arrays (FPGAs). His Ph.D. degree was devoted to introducing new MoM/GA-based algorithms for different digital beamforming applications which are of main concern in the recent and future wireless communication systems. He was an Associate Professor with Tanta University, in 2017. He has also introduced VHDL implementations of both DOA estimation algorithm and the fixed beamwidth electronic scanning (FBWES) algorithm. Furthermore, he has introduced several image encryption and compression techniques for efficient images handling.