IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 29, 2018, accepted July 27, 2018, date of publication August 20, 2018, date of current version September 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2866089

Sparsity-Based Image Inpainting Detection via Canonical Correlation Analysis With Low-Rank Constraints

XIAO JIN[1,2], (Student Member, IEEE), YUTING SU[1], LIANG ZOU[3,4], YONGWEI WANG[2], PEIGUANG JING[1], AND Z. JANE WANG[2], (Fellow, IEEE)

[1]School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China
[2]Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC V6T 1Z4, Canada
[3]School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798
[4]Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly, Nanyang Technological University, Singapore 639798

Corresponding author: Yuting Su (ytsu@tju.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61572356 and Grant 61303208 and in part by the Tianjin Research Program of Application Foundation and Advanced Technology under Grant 15JCQNJC41600.

ABSTRACT Image inpainting, a commonly used image editing technique for filling the mask or missing areas in images, is often adopted to destroy the integrity of images by forgers with ulterior motives. Compared with the other types of inpainting, the sparsity-based inpainting exploits more general prior knowledge and has a broader application scope. Although many methods for detecting exemplar-based and diffusion-based inpainting have been successfully studied in the literature, there is still a lack of effective schemes for detecting the sparsity-based inpainting. In this paper, to fill this gap, we proposed a novel algorithm for sparsity-based image inpainting detection. We revealed the potential connection between sparsity-based inpainting and canonical correlation analysis (CCA). This type of inpainting has a strong effect on the CCA coefficients. Based on this observation, a modified objective function of CCA and a corresponding optimization algorithm are further proposed to enhance the inter-class difference in our feature set. Experimental results on three publicly available data sets demonstrated our method's superiority over other competitors. Particularly, compared with previous inpainting detection methods, the proposed framework yields better performances in the cases of JPEG compression and Gaussian noise addition. The proposed method also shows promising results when employed to detect other types of inpainting.

INDEX TERMS Image forensics, image inpainting detection, sparse learning, canonical correlation analysis (CCA).

I. INTRODUCTION

During the past decade, many image editing software, which is convenient and labor-light, has been made available publicly. Although such editing tools are not designed for tampering images with vicious intentions, non-experts are able to produce a forged image without noticeable artifact. To fight against the malicious forgers, image forensics has been emerging [1]–[3]. Most of the forensic tasks are focused on one or several specific tampering behaviours, *e.g.* resampling detection [4], [5], copy-move detection [6], [7], camera model identification [8], [9], JPEG compression [10], [11], spatial filtering [12], [13], and contrast enhancement [14], [15].

For a particular type of forgery action, the judicial authority needs to develop novel forensic schemes to defend it. In recent years, many types of image inpainting algorithms have been proposed, *e.g.* GAN-based methods [16], low-rank based methods [17], diffusion-based methods [18], exemplar-based methods [19], and sparsity-based methods [20]. While image inpainting has been drawing increasing research attention for years, the corresponding image inpainting detection method has attracted relatively much less attention. Although malicious tampering is not the original intention of image inpainting technology, forgers can easily get and use these image inpainting software to tamper some images with ulterior motives, *e.g.* object removal [19]. Also inpainting is ordinarily used as post-processing to retouch the traces left by other tampering operations. If an image is identified as an inpainted one, this image can be highly suspicious.

Researchers investigated the traces of exemplar-based inpainting for image forensics. Wu *et al.* [21] introduced the zero-connectivity features to calculate the matching degree of test blocks, and made a distinction between original regions and inpainted regions with the help of fuzzy membership. Bacchuwar *et al.* [22] exploited the luminance component of the image and median comparison of the blocks to propose the "Jump Patch Block Match" algorithm. It greatly reduced the computational cost. Chang *et al.* [23] searched similar blocks in an image and removed the false positives caused by uniform areas. Their automatic forgery detection framework is terminated by identifying forged areas with the multi-region relation algorithm. Liang *et al.* [24] integrated central pixel mapping, greatest zero-connectivity component labeling and fragment splicing detection into a joint detection framework with high efficiency. Zhang *et al.* [25] indicated that post-processing, *e.g.* JPEG compression, Gaussian noise addition, destructs the similarity among block pairs and synchronously disorders the correlations of adjacent pixels. It was shown that using the joint probability density matrix of DCT coefficients as a feature set can achieve high performance with post-processing.

Li *et al.* [26] aimed to detect image inpainting in the JPEG format. They used the first digits of individual alternate current modes to reveal the different JPEG compression history in different image patches. Zhao *et al.* [27] designed the feature set by using absolute differences between the test image and the re-saved images with different JPEG quality factors. Diffusion-based inpainting detection has been recently studied by Li *et al.* [28]. The authors reviewed the process of this type of inpainting and found that it tends to preserve the isophote direction. A feature set depending on the changes in the image Laplacians along the direction of isophote is extracted. Experimental results show high performance even under interferences of gamma correction, rotation, scaling and JPEG compression.

Although some pioneers have made many worthwhile contributions to detect diffusion-based and exemplar-based inpainting, these two types of image inpainting have limitations in real-world applications. Diffusion-based inpainting generally can obtain satisfied results in small regions, however it may cause some blurring artifacts, especially when missing parts of the image are large or complex [29], [30]. This type of inpainting tends to reconstruct flat-looking images due to the lack of semantic texture/structure synthesis [31]. It performs poorly when adapted to process images that are continuous by parts (cartoon-like) [32]. Exemplar-based inpainting methods usually can manipulate large masks in an image, however they may fail when there is no strong self-similarity within target images [33]. It is also likely to cause discontinuous edges along the patches since these methods generally do not check for visual consistency [34].

Apart from diffusion-based and exemplar-based methods, sparsity-based inpainting is another complementary type of inpainting, though it is largely ignored by researchers. Compared with the above two types of methods, sparsity-based methods use more general prior knowledge in image processing and also have a broader scope of applications. It is shown that sparsity-based inpainting is more robust to noise [35] and more suitable for filling large texture areas [36]. Especially in the application of missing block completion, sparsity-based methods can fill in the missing region with relatively composite textures and structures effectively [31]. In this paper, we focus on the detection of sparsity-based image inpainting, a largely ignored research topic in the current literature.

To better distinguish the forgery image edited by sparsity-based inpainting, we propose a novel method to detect this type of inpainting. We study the changes of signal linear dependence after the inpainting operation. Motivated by these informative changes, a modified CCA objective function is proposed to model the changes, and a corresponding optimization algorithm is proposed by using the Alternating Direction Method of Multipliers (ADMM). Experiments show competitive and robust results with different parameters. The main contributions of this paper can be summarized as follows:

1) We review the fundamental principles of sparsity-based inpainting to identify the hidden traces in images created by this operation. To the best of our knowledge, it is the first work to detect sparsity-based inpainting.
2) We uncover the relationship between sparsity-based inpainting and the traditional CCA. To characterize inpainting signals properly, a modified CCA formulation is presented. We also develop an iterative algorithm for optimizing the new objective function by ADMM.
3) Experimental results on several public datasets show that the proposed method achieves significant performance improvements. The proposed method shows its robustness for JPEG compression and Gaussian noise.

The remainder of this paper is organized as follows. Section II reviews the procedure of sparsity-based image inpainting and CCA, and illustrates the inpainting effects on coefficients of CCA. In section III, we formulate the objective function and present the optimization algorithm. Extensive experimental results with various parameters are presented in section IV. Finally, we conclude our paper and outline the future work in section V.

II. BACKGROUND

A. SPARSITY-BASED IMAGE INPAINTING

Image sparsity priors have attracted research attention for years in the image processing community. In [20], Elad pointed out that the image inpainting problem can also be handled assuming sparsity priors. Generally, a block of an image can be vectorized into $\mathbf{z} \in \mathbb{R}^{d_1}$. Assuming that $\mathbf{z}$ can be represented by a predefined redundant dictionary $\mathbf{D} \in \mathbb{R}^{d_1 \times d_2}$ and a sparse vector $\alpha \in \mathbb{R}^{d_1}$, such as $\|\alpha\|_0 = k_0$, $\mathbf{z} = \mathbf{D}\alpha$.

We then define a degradation matrix $\mathbf{M} \in \mathbb{R}^{(p-q)\times p}$ that removes q samples from the signal $\mathbf{z}$. Corresponding to the mask positions, $\mathbf{M}$ is built by removing arbitrary q rows from

(a) (b) (c)

FIGURE 1. An example of sparsity-based image inpainting is presented. (a) an original block. (b) a block with a mask. (c) an inpainted block.

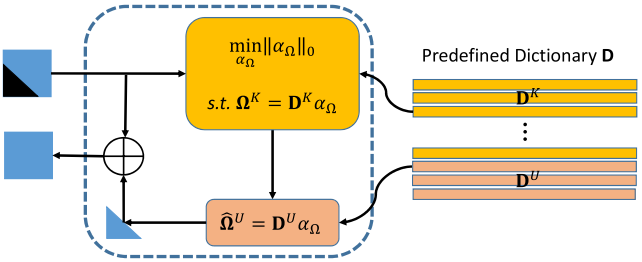


FIGURE 2. An illustration of sparsity-based image inpainting. There are two major steps in sparsity-based image inpainting process. (1) Use the original signals and a part of the predefined dictionary to learn a sparse vector from sparse learning; and (2) Use the sparse vector and another part of the predefined dictionary to learn the inpainting signals. It is assumed in sparsity-based inpainting that both known and unknown regions share the same sparsity in signals. Therefore, they have the same sparse vector α_Ω in the sparse learning process.

a $p \times p$ identity matrix. The image inpainting problem can be formulated as:

$$\min_{\alpha} \|\alpha\|_0 \quad s.t.\ \mathbf{z} = \mathbf{MD}\alpha. \tag{1}$$

In the real inpainting scenarios, the process has two major steps. The sparse vector α_Ω is first learned by the known part pixels Ω^K:

$$\min_{\alpha_\Omega} \|\alpha_\Omega\|_0 \quad s.t.\ \Omega^K = \mathbf{D}^K\alpha_\Omega, \tag{2}$$

where $\mathbf{D}^K = \mathbf{MD}$ removes the elements of $\mathbf{D}$ in each masked position. Thereafter, the unknown part pixels Ω^U can be estimated as:

$$\hat{\Omega}^U = \mathbf{D}^U\alpha_\Omega, \tag{3}$$

where $\mathbf{D}^U$ represents the elements in $\mathbf{D}$ corresponding to the positions of unknown pixels.

This type of inpainting mainly exploits the sparse priors in image processing. The general assumption in sparsity-based image inpainting is that both the known and unknown regions share the same sparsity in signals. Thus, they have the same sparse vector α_Ω in the sparse learning process. Fig. 2 illustrates the process of sparsity-based image inpainting.

Compared with other types of image inpainting operations, *e.g.* diffusion-based inpainting and exemplar-based inpainting, sparsity-based inpainting uses more general prior knowledge in image processing. Consequently, it also has a wider range of practical applications. No matter in flat image regions or structured texture areas, sparse learning can preserve signal characteristics appropriately.

B. CANONICAL CORRELATION ANALYSIS (CCA)

As a result of sparsity-based inpainting, the inpainted signals are relevant to the same redundant dictionary. This fact inspired us to uncover the hiding traces of inpainted signals via CCA that is a measurement of the relationship between two variables. Suppose we have $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ and $\mathbf{Y} \in \mathbb{R}^{n_1 \times n_2}$ with covariances Σ_{XX}, Σ_{YY} and cross-covariance Σ_{XY} respectively. The purpose of CCA is to find two bases $\{\mathbf{A}, \mathbf{B}\}$ to have the maximal correlation between linear combinations $\mathbf{A}^T\mathbf{X}$ and $\mathbf{B}^T\mathbf{Y}$:

$$\begin{aligned}\{\mathbf{A}, \mathbf{B}\} &= \arg\max_{\mathbf{A},\mathbf{B}} corr(\mathbf{A}^T\mathbf{X}; \mathbf{B}^T\mathbf{Y}) \\ &= \arg\max_{\mathbf{A},\mathbf{B}} \frac{\mathbf{A}^T\Sigma_{XY}\mathbf{B}}{\sqrt{\mathbf{A}^T\Sigma_{XX}\mathbf{A}\mathbf{B}^T\Sigma_{YY}\mathbf{B}}}.\end{aligned} \tag{4}$$

To solve the above objective function, we add the same constraints as those in [37]. In this way, it can be simplified as:

$$\begin{aligned}&\max_{\mathbf{A},\mathbf{B}} tr(\mathbf{A}^T\Sigma_{XY}\mathbf{B}) \\ &s.t.\ \mathbf{A}^T\Sigma_{XX}\mathbf{A} = \mathbf{B}^T\Sigma_{YY}\mathbf{B} = \mathbf{I},\end{aligned} \tag{5}$$

where $tr(\cdot)$ indicates the trace of a matrix and $\mathbf{I}$ means the identity matrix.

C. ARTIFACTS OF SPARSITY-BASED IMAGE INPAINTING ON CCA COEFFICIENTS

As illustrated in Fig. 3, there is a potential connection between sparsity-based image inpainting and CCA. They share the similar mathematical formulation and assumption.

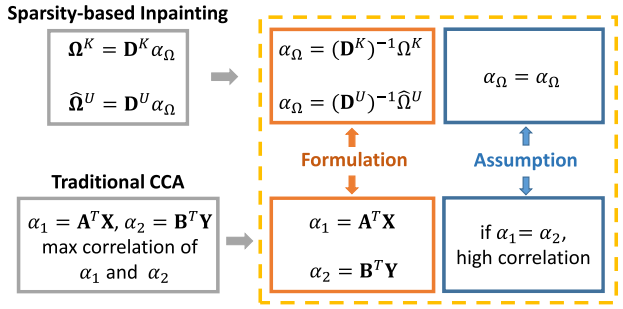


FIGURE 3. The potential connection between sparsity-based image inpainting and CCA. Their mathematical formulation and assumption are similar: (1) They are both matrix projection in a mathematical formulation; and (2) Sparsity-based inpainting assumes that the sparse vectors are equal for untouched signals and inpainting signals, while canonical correlation analysis assumes that the canonical variables have maximum correlations. If the canonical variates are equal, they will have high correlations.

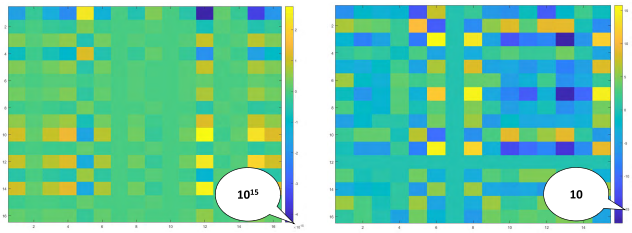


FIGURE 4. Illustrative effects of the sparsity-based image inpainting operation on traditional CCA coefficients. We note significant differences in the magnitude of absolute values after the sparsity-based image inpainting operation. (a) CCA coefficients of an original image block; and (b) CCA coefficients of an inpainted image block.

They are both matrix projection in a mathematical formulation. Sparsity-based inpainting assumes that the sparse vectors are equal for untouched signals and inpainting signals, while CCA assumes that the canonical variables have the maximum correlations. If the canonical variables are highly similar, they should have high correlations.

According to [20], in a practical sparsity-based inpainting process, if a randomly selected row is discarded in $\mathbf{D}$, it may cause changes in linear dependency. The coefficients in CCA can reveal the linear relationship in the data. To this end, we apply the traditional CCA on inpainted image blocks, and the results are illustrated in Fig. 4. As we can see in the figure, sparsity-based image inpainting significantly changes the absolute values of the basis in CCA after inpainting.

Furthermore, in the view of CCA, we can use the *rank* of $\mathbf{A}^T$ and $\mathbf{B}^T$ to represent the linear dependence of $\mathbf{D}^K$ and $\mathbf{D}^U$ in inpainting process. From this point of view, the *rank* of the input data or the basis in CCA can be an effective feature set to distinguish the inpainted signals. We therefore further propose a modified CCA formulation with low-rank constraints. Our scheme will be presented in the following section. By adjusting the parameter balancing the traditional CCA item and the low-rank regularization item, we can obtain an effective feature set.

III. THE PROPOSED METHOD

In this section, we describe the proposed method in detail. First, we define a modified CCA objective function with low-rank constraints. Further, an optimization algorithm is proposed to solve it.

A. FORMULATION OF CCA WITH LOW-RANK CONSTRAINTS (CCALR)

Inspired by the observations in the previous section, we attempt to add constraints on the coefficient of the traditional CCA to improve the classification performance. Hence, we define the following objective function with low-rank constraints on $\mathbf{A}$ and $\mathbf{B}$:

$$\begin{aligned} &\min_{\mathbf{A},\mathbf{B}} \ rank(\mathbf{A}) + rank(\mathbf{B}) - \delta tr(\mathbf{A}^T \Sigma_{XY} \mathbf{B}) \\ &s.t.\ \mathbf{A}^T \Sigma_{XX} \mathbf{A} = \mathbf{B}^T \Sigma_{YY} \mathbf{B} = \mathbf{I}, \end{aligned} \tag{6}$$

where $rank(\cdot)$ is the rank operator of a matrix, δ is a predefined balance parameter. According to Sylvester's inequality [38], if $\{\mathbf{A}, \mathbf{B}\} \in \mathbb{R}^{m_1 \times m_2}$, $m = \min\{m_1, m_2\}$,

$$rank(\mathbf{A}) + rank(\mathbf{B}) - m \leqslant \min\{rank(\mathbf{A}), rank(\mathbf{B})\}. \tag{7}$$

Without loss of generality, assuming $rank(\mathbf{A}) \leqslant rank(\mathbf{B})$, we can get

$$\begin{aligned} &\min_{\mathbf{A},\mathbf{B}} \ rank(\mathbf{A}) - \delta tr(\mathbf{A}^T \Sigma_{XY} \mathbf{B}) \\ &s.t.\ \mathbf{A}^T \Sigma_{XX} \mathbf{A} = \mathbf{B}^T \Sigma_{YY} \mathbf{B} = \mathbf{I}. \end{aligned} \tag{8}$$

Since Eq.(8) is discontinuous and non-convex because of the $rank(\cdot)$ operation, it is hard to find an optimal solution. As a common practice in rank minimization problems, we relax it to the nuclear norm $\|\cdot\|_*$ which is the sum of singular values of a matrix. In addition, we also relax the constraint of $\mathbf{A}^T \Sigma_{XX} \mathbf{A} = \mathbf{I}$ in traditional CCA. Thus, our objective function becomes a convex optimization problem:

$$\begin{aligned} &\min_{\mathbf{A},\mathbf{B}} \ \|\mathbf{A}\|_* - \delta tr(\mathbf{A}^T \Sigma_{XY} \mathbf{B}) \\ &s.t.\ \mathbf{B}^T \Sigma_{YY} \mathbf{B} = \mathbf{I}. \end{aligned} \tag{9}$$

In this format of our objective function, since both the traditional CCA item $\delta tr(\mathbf{A}^T \Sigma_{XY} \mathbf{B})$ and the low-rank regularization item $\|\mathbf{A}\|_*$ are convex, the whole function in Eq.(9) is also convex [39]–[41]. In the following, we will introduce an optimization scheme to solve Eq.(9) by ADMM.

B. OPTIMIZATION

In this part, we adopt ADMM [42] to solve the objective function Eq.(9). ADMM divides a complicated problem into a couple of separable subproblems, which is widely used in low-rank optimization.

First, as a common approach for previous solutions of low-rank optimization, we introduce a relaxed variable $\mathbf{C}$ to substitute $\mathbf{A}$, reformulating Eq.(9) into:

$$\begin{aligned} &\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \ \|\mathbf{A}\|_* - \delta tr(\mathbf{C}^T \Sigma_{XY} \mathbf{B}) \\ &s.t.\ \mathbf{B}^T \Sigma_{YY} \mathbf{B} = \mathbf{I}, \mathbf{A} = \mathbf{C}. \end{aligned} \tag{10}$$

The augmented Lagrangian function of Eq.(10) can be written as $\mathcal{L}$:

$$\begin{aligned}\mathcal{L}(\mathbf{A},\mathbf{B},\mathbf{C},\mathbf{Y}_1,\mathbf{Y}_2,\mu)&=\|\mathbf{A}\|_*-\delta tr(\mathbf{C}^T\Sigma_{XY}\mathbf{B})\\&+\left\langle\mathbf{Y}_1,\mathbf{B}^T\Sigma_{YY}\mathbf{B}-\mathbf{I}\right\rangle+\langle\mathbf{Y}_2,\mathbf{A}-\mathbf{C}\rangle\\&+\frac{\mu}{2}\left(\left\|\mathbf{B}^T\Sigma_{YY}\mathbf{B}-\mathbf{I}\right\|_F^2+\|\mathbf{A}-\mathbf{C}\|_F^2\right),\\&=\|\mathbf{A}\|_*-\delta tr(\mathbf{C}^T\Sigma_{XY}\mathbf{B})\\&+\frac{1}{2}\left\|\mathbf{B}^T\Sigma_{YY}\mathbf{B}-\mathbf{I}+\frac{\mathbf{Y}_1}{\mu}\right\|_F^2+\frac{1}{2}\left\|\mathbf{A}-\mathbf{C}+\frac{\mathbf{Y}_2}{\mu}\right\|_F^2,\end{aligned}\tag{11}$$

where $\langle\cdot,\cdot\rangle$ denotes the trace of inner product, $\|\cdot\|_F$ is the Frobenius norm, $\mathbf{Y}_1$ and $\mathbf{Y}_2$ are two Lagrange multipliers, μ is a penalty parameter.

For better explanation, we use t to indicate the tth iteration step in ADMM. $\mathbf{A}_t, \mathbf{B}_t, \mathbf{C}_t, \mathbf{Y}_{1,t}, \mathbf{Y}_{2,t}$ and μ_t are defined as the variables updated in the tth iteration. Hence, each variable in the $t+1$ iteration can be calculated as follows:

(1) **For A**:

By fixing $\mathbf{B}_t, \mathbf{C}_t, \mathbf{Y}_{1,t}, \mathbf{Y}_{2,t}, \mu_t$ and eliminating irrelevant terms, $\mathbf{A}_{t+1}$ can be optimized by:

$$\begin{aligned}\mathbf{A}_{t+1}&=\arg\min_{\mathbf{A}}\mathcal{L}(\mathbf{A},\mathbf{B}_t,\mathbf{C}_t,\mathbf{Y}_{1,t},\mathbf{Y}_{2,t},\mu_t)\\&=\arg\min_{\mathbf{A}}\frac{1}{2}\left\|\mathbf{A}-(\mathbf{C}_t-\frac{\mathbf{Y}_{2,t}}{\mu_t})\right\|_F^2+\|\mathbf{A}\|_*.\end{aligned}\tag{12}$$

Since Eq.(12) is strictly convex, there exists a unique solution. It is also a standard form of the Singular Value Thresholding Algorithm [43] with a closed form solution. Under the framework in [43], we first define $\Phi=\mathbf{C}_t-\mathbf{Y}_{2,t}/\mu_t$ and perform the Singular Value Decomposition (SVD) as $\Phi=\mathbf{U}\Theta\mathbf{V}^T$, $\Theta=diag(\{\sigma_i\}_{i=1}^r)$. The matrices $\mathbf{U}$ and $\mathbf{V}$ are left-singular and right-singular matrices respectively, and r is the rank of Φ. Accordingly the optimal $\mathbf{A}_{t+1}$ can be updated as:

$$\mathbf{A}_{t+1}=\mathcal{D}_1(\Phi),\tag{13}$$

where the singular value shrinkage operator is utilized by $\mathcal{D}_1(\Phi)=\mathbf{U}\mathcal{D}_1(\Theta)\mathbf{V}^T$, $\mathcal{D}_1(\Theta)=diag(\{\sigma_i-1\}_+)$ and the subscript "+" denotes the positive part of $\{\sigma_i-1\}$.

(2) **For B**:

While keeping the variables $\mathbf{A}_t, \mathbf{C}_t, \mathbf{Y}_{1,t}, \mathbf{Y}_{2,t}, \mu_t$ fixed, the optimization of $\mathbf{B}_{t+1}$ can be reformulated by:

$$\begin{aligned}\mathbf{B}_{t+1}&=\arg\min_{\mathbf{B}}\mathcal{L}(\mathbf{A}_t,\mathbf{B},\mathbf{C}_t,\mathbf{Y}_{1,t},\mathbf{Y}_{2,t},\mu_t)\\&=\arg\min_{\mathbf{B}}\{-\delta tr(\mathbf{C}_t^T\Sigma_{XY}\mathbf{B})\\&+\frac{1}{2}\left\|\mathbf{B}^T\Sigma_{YY}\mathbf{B}-\mathbf{I}+\frac{\mathbf{Y}_{1,t}}{\mu_t}\right\|_F^2\}.\end{aligned}\tag{14}$$

To solve Eq.(14), we calculate the derivative of Eq.(11) with respect to $\mathbf{B}$, and obtain:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{B}}=\Sigma_{YY}\mathbf{B}\frac{\mathbf{Y}_{1,t}^T}{\mu_t}+\Sigma_{YY}^T\mathbf{B}\frac{\mathbf{Y}_{1,t}}{\mu_t}-\delta\Sigma_{XY}^T\mathbf{C}_t.\tag{15}$$

By setting this partial derivative of Eq.(11) to be zero, $\mathbf{B}_{t+1}$ can be updated as:

$$\mathbf{B}_{t+1}=\delta\mu_t\Sigma_{YY}^{-1}\Sigma_{XY}^T\mathbf{C}_t\cdot\left(\mathbf{Y}_{1,t}^T+\mathbf{Y}_{1,t}\right)^{-1}.\tag{16}$$

(3) **For C**:

Similarly, by ignoring the terms independent of $\mathbf{C}$ in Eq.(11), the solution of $\mathbf{C}_{t+1}$ can be rewritten as:

$$\begin{aligned}\mathbf{C}_{t+1}&=\arg\min_{\mathbf{C}}\mathcal{L}(\mathbf{A}_t,\mathbf{B}_t,\mathbf{C},\mathbf{Y}_{1,t},\mathbf{Y}_{2,t},\mu_t)\\&=\arg\min_{\mathbf{C}}\{-\delta tr(\mathbf{C}^T\Sigma_{XY}\mathbf{B}_t)\\&+\frac{1}{2}\left\|\mathbf{A}_t-\mathbf{C}+\frac{\mathbf{Y}_{2,t}}{\mu_t}\right\|_F^2\}.\end{aligned}\tag{17}$$

Dropping the constant terms, the derivative of Eq.(11) with respect to $\mathbf{C}$ is computed as:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{C}}=\mathbf{C}-\mathbf{A}_t-\frac{\mathbf{Y}_{2,t}}{\mu_t}-\delta\Sigma_{XY}\mathbf{B}_t.\tag{18}$$

Setting this partial derivative of Eq.(11) to be zero, we can derive:

$$\mathbf{C}_{t+1}=\mathbf{A}_t+\frac{\mathbf{Y}_{2,t}}{\mu_t}+\delta\Sigma_{XY}\mathbf{B}_t.\tag{19}$$

(4) **For $\mathbf{Y_1}$, $\mathbf{Y_2}$ and μ**:

Finally, the Lagrange multipliers $\mathbf{Y}_1$, $\mathbf{Y}_2$ and penalty parameter μ can be updated as:

$$\begin{aligned}\mathbf{Y}_{1,t+1}&=\mathbf{Y}_{1,t}+\mu_t\left(\mathbf{B}_t^T\Sigma_{YY}\mathbf{B}_t-\mathbf{I}\right)\\\mathbf{Y}_{2,t+1}&=\mathbf{Y}_{2,t}+\mu_t\left(\mathbf{A}_t-\mathbf{C}_t\right)\\\mu_{t+1}&=\min(\rho\mu_t,\mu_{max}),\end{aligned}\tag{20}$$

where μ_{max} is the upper bound of μ, and ρ is a constant chosen in advance.

To clearly describe the above proposed framework (e.g., the initialization step), its main steps are given in Algorithm 1.

IV. EXPERIMENTAL RESULTS

In this section, we carry out experiments under different circumstances to demonstrate the performance of the proposed method. We give a detailed description of datasets and the experimental setup. Then convergence and parameter sensitivity are taken into consideration. Finally, the performance of synthetic inpainted images with and without post-processing is shown.

A. DATASETS AND EXPERIMENTAL SETUP

In this paper, we evaluate different methods on three widely used forensic datasets:

- **Uncompressed Color Image Database (UCID)** [44]: It is a commonly used dataset which consists

Algorithm 1 The Proposed Algorithm

Require:
Inpainted signals $\mathbf{X}$, $\mathbf{Y}$, the balance parameter δ;
Ensure:
The bases of CCALR, $\mathbf{A}$, $\mathbf{B}$;
1: Initialize $t = 0$, $\mathbf{A}_0$ and $\mathbf{B}_0$ as two random matrices, $\mathbf{C}_0 = \mathbf{A}_0$, $\mathbf{Y}_{1,0} = \mathbf{Y}_{2,0} = \mathbf{0}$;
2: Initialize $\rho = 1.9$, $\mu_0 = 10^{-3}$, $\mu_{max} = 10^6$;
3: **while** not converged **do**
4: Fix others and update $\mathbf{A}_{t+1}$ by Eq.(13): $\mathbf{A}_{t+1} \leftarrow \mathcal{D}_1(\Phi)$;
5: Fix others and update $\mathbf{B}_{t+1}$ by Eq.(16): $\mathbf{B}_{t+1} \leftarrow \delta\mu_t \Sigma_{YY}^{-1}\Sigma_{XY}^T\mathbf{C}_t \cdot \left(\mathbf{Y}_{1,t}^T + \mathbf{Y}_{1,t}\right)^{-1}$;
6: Fix others and update $\mathbf{C}_{t+1}$ by Eq.(19): $\mathbf{C}_{t+1} \leftarrow \mathbf{A}_t + \mathbf{Y}_{2,t}/\mu_t + \delta\Sigma_{XY}\mathbf{B}_t$;
7: Update the multipliers $\mathbf{Y}_{1,t+1}$ and $\mathbf{Y}_{2,t+1}$ by Eq.(20): $\mathbf{Y}_{1,t+1} \leftarrow \mathbf{Y}_{1,t} + \mu_t\left(\mathbf{B}_t^T\Sigma_{YY}\mathbf{B}_t - \mathbf{I}\right)$ $\mathbf{Y}_{2,t+1} \leftarrow \mathbf{Y}_{2,t} + \mu_t\left(\mathbf{A}_t - \mathbf{C}_t\right)$;
8: Update the parameter μ_{t+1} by: $\mu_{t+1} \leftarrow \min(\rho\mu_t, \mu_{max})$;
9: Check the convergence conditions;
10: Update the iteration variable using $t = t + 1$;
11: **end while**

of 1,338 uncompressed TIFF images on a variety of topics including natural scenes and man-made objects, both indoors, and outdoors.[1] The typical image sizes are 512 × 384 or 384 × 512. We use this dataset to demonstrate the effectiveness of our method in low-resolution formats.

- **Raw Images Dataset (RAISE)** [45]: It is a new collection of untouched and diverse data which covers 8,156 raw images including a wide variety of both semantic contents and technical parameters.[2] We conduct experiments on this dataset to show our method's performance on high-resolution images.
- **Dresden Image Database (DID)** [46]: It is a traditional dataset for forensics with more than 14,000 images covering different camera settings, environments and specific scenes.[3] This dataset is often used for camera identification by many previous researchers since it contains images captured by various camera brands, models, and settings. We use this dataset to show the reliability of our method for different camera models.

In [20], Elad summarized the fundamental theory of sparsity-based image inpainting and provided open source codes for implementing his sparsity-based type of inpainting. Compared with other work of sparsity-based inpainting, this work [20] is more general and more suitable for images with moderate missing regions. Detection of the work [20] is meaningful in practical applications. In the experiments, we use the codes in Elad's work [20] for inpainting the images.[4] An example of an inpainted image by his code is shown in Fig. 1.

Given an RGB color image, the test image is cut into blocks with size b, *e.g.* 32 × 32, 64 × 64, and then mask locations are chosen randomly. We repeat the same inpainteing process for three RGB channels respectively and then merge them together to create an inpainted RGB image. As Fig. 1 shows, the sparsity-based inpainting can fill the mask perfectly when the dictionary is redundant and the reconstruction error σ is selected properly.

The left and right parts of a test block are taken as the input matrices $\mathbf{X}$ and $\mathbf{Y}$ in our method. Specifically, for each color channel, we use the `im2col()` function in Matlab to rearrange each distinct 4-by-4 patch in the left or right part of the input block into a column of 16 elements. Then, we concatenate all three channels as columns of the matrices $\mathbf{X}$ and $\mathbf{Y}$. Finally, we obtain our input matrices $\{\mathbf{X}, \mathbf{Y}\} \in \mathbb{R}^{16\times(3b^2/16)}$. In the view of CCA, 16 denotes the number of observations (rows), while $(3b^2/16)$ represents the number of variables (columns).

In our preliminary experiments, the CCA coefficients and the *rank* of these coefficients are extracted as the feature set separately. We find that both of them are effective to distinguish the inpainted and original blocks. In the following results, a test block is classified as inpainted when it is classified as inpainted by both the CCA coefficients and the *rank* of these coefficients.

Here the widely used libSVM [47] is employed for classification. The performance is generally evaluated by the following three criteria, where TP (True Positive), FP (False Positive), FN (False Negative), TN (True Negative) represent the corresponding number of blocks respectively:

- **Accuracy**: A measure indicates the fraction of true samples which are classified correctly among all samples.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \tag{21}$$

- **Recall (or Sensitivity)**: A measure denotes the proportion of correctly detected positive results among all positive results.

$$Recall = \frac{TP}{TP + FN}. \tag{22}$$

- **F1-score**: A measure that combines precision and recall is the harmonic mean of precision and recall.

$$F1 = \frac{2TP}{2TP + FN + FP}. \tag{23}$$

Our experiments were implemented on the machine which includes 16 GB of RAM and an Intel Core i7-3770 CPU. The average results of 5 trials are presented as the final experimental results.

[1] http://vision.doc.ntu.ac.uk/
[2] http://mmlab.science.unitn.it/RAISE/
[3] http://forensics.inf.tu-dresden.de/ddimgdb/
[4] Matlab codes and supplementary materials for his work are available at http://www.cs.technion.ac.il/~elad/software/

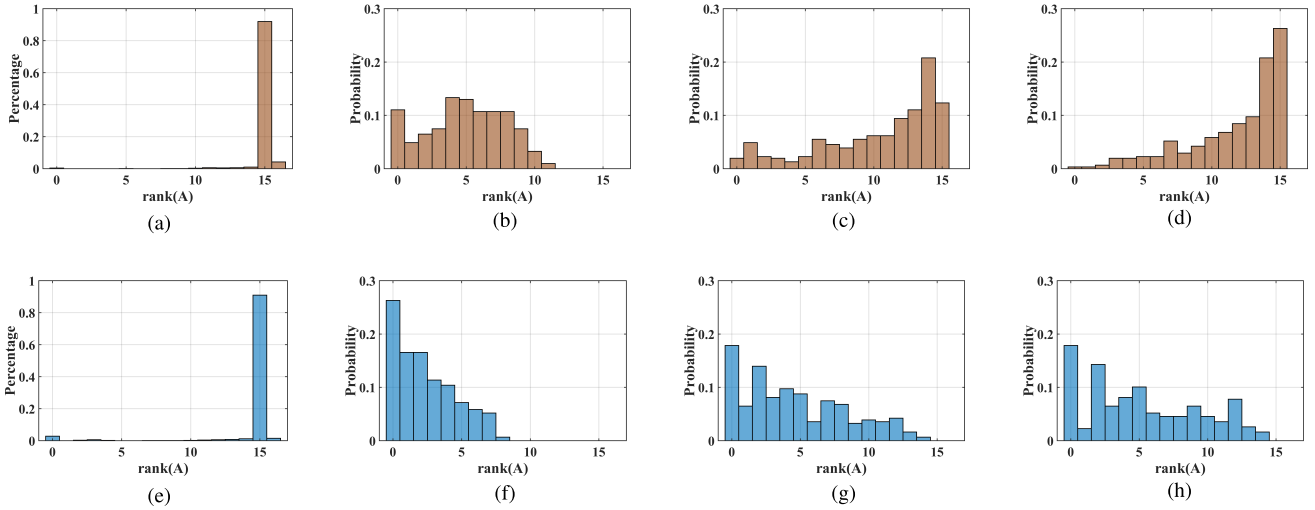


FIGURE 5. Histogram distributions of *rank*(**A**) corresponding to original blocks and tampered blocks respectively. We collected 2,500 original and inpainted blocks of size 32 × 32. The first row represent the original data case, while the second row represents the inpainted data. As the parameter δ changes, there are significant differences between them. (a) original blocks, traditional CCA. (b) original blocks, δ=10. (c) original blocks, δ=100. (d) original blocks, δ=200. (e) inpainted blocks, traditional CCA. (f) inpainted blocks, δ=10. (g) inpainted blocks, δ=100. (h) inpainted blocks, δ=200.

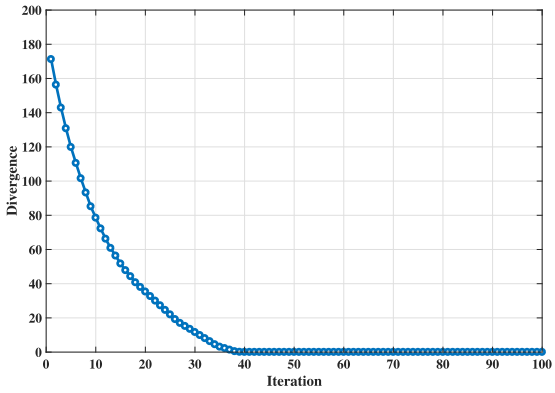


FIGURE 6. A typical convergence curve of our proposed method. The horizontal axis represents the index of iterations and the vertical axis is the divergence between two consecutive objective function values.

B. PERFORMANCE EVALUATION

Before reporting the detection performances on inpainted images, we evaluate our objective function and optimization algorithm in terms of convergence, parameter, and inter-class discrimination.

1) EVALUATION ON CONVERGENCE

In this part, we evaluate the convergence of our objective function and randomly select a trial to report the results. To guarantee that the final results are steady, we define the difference of the objective functions from two consecutive iterations as the convergence condition:

$$D_t = |F_t - F_{t-1}|, \tag{24}$$

where F_t and F_{t-1} represent the objective functions at the t and $t+1$ iterations, respectively. Fig. 6 shows the absolute value of the difference at different iterations. As the index of the iteration increases, the value decreases rapidly. Based on previous studies, it is critical to choose an appropriate convergence condition for an optimization problem. We set the stopping criteria as follows: the relative change between two iterations is below the threshold of 1e-6 and the maximum number of iterations is 100.

2) PARAMETER SENSITIVITY ANALYSIS

In this subsection, we conduct experiments to investigate the effect of the balance parameter δ. As shown in Eq.(9), δ is used for balancing the traditional CCA item and the low-rank regularization item. An appropriate parameter has a great effect on final results. The histograms of *rank*(**A**) are taken as an example in Fig 5. The horizontal axis represents the rank of data, and the vertical axis is the relative probability. Brown represents the original data, while blue stands for inpainted data. Under different parameters, the extracted features have different discriminative characteristics.

To find a good parameter δ, we select 2,500 image blocks from UCID, RAISE and DID respectively. The block size is set to be 32 × 32. Detection accuracy is shown in Table 1 with various δ. Although images from different datasets have different resolutions and sensor noises, the detection algorithm has the best performance when δ equals to 200, among all datasets. Notably, in this preliminary experiment, the results of our proposed method in RAISE, a dataset with higher resolution, is better than that in other datasets.

C. PERFORMANCE OF SYNTHETIC INPAINTED IMAGES

In this experiment, 100 images are selected from the above three datasets respectively. Specifically, the central parts are selected and then divided into 9 blocks (3 × 3). The sparsity-based inpainting operation [20] is implemented to those blocks. Finally, we obtain 900 positive samples and 900 negative samples in each dataset. In our experimental

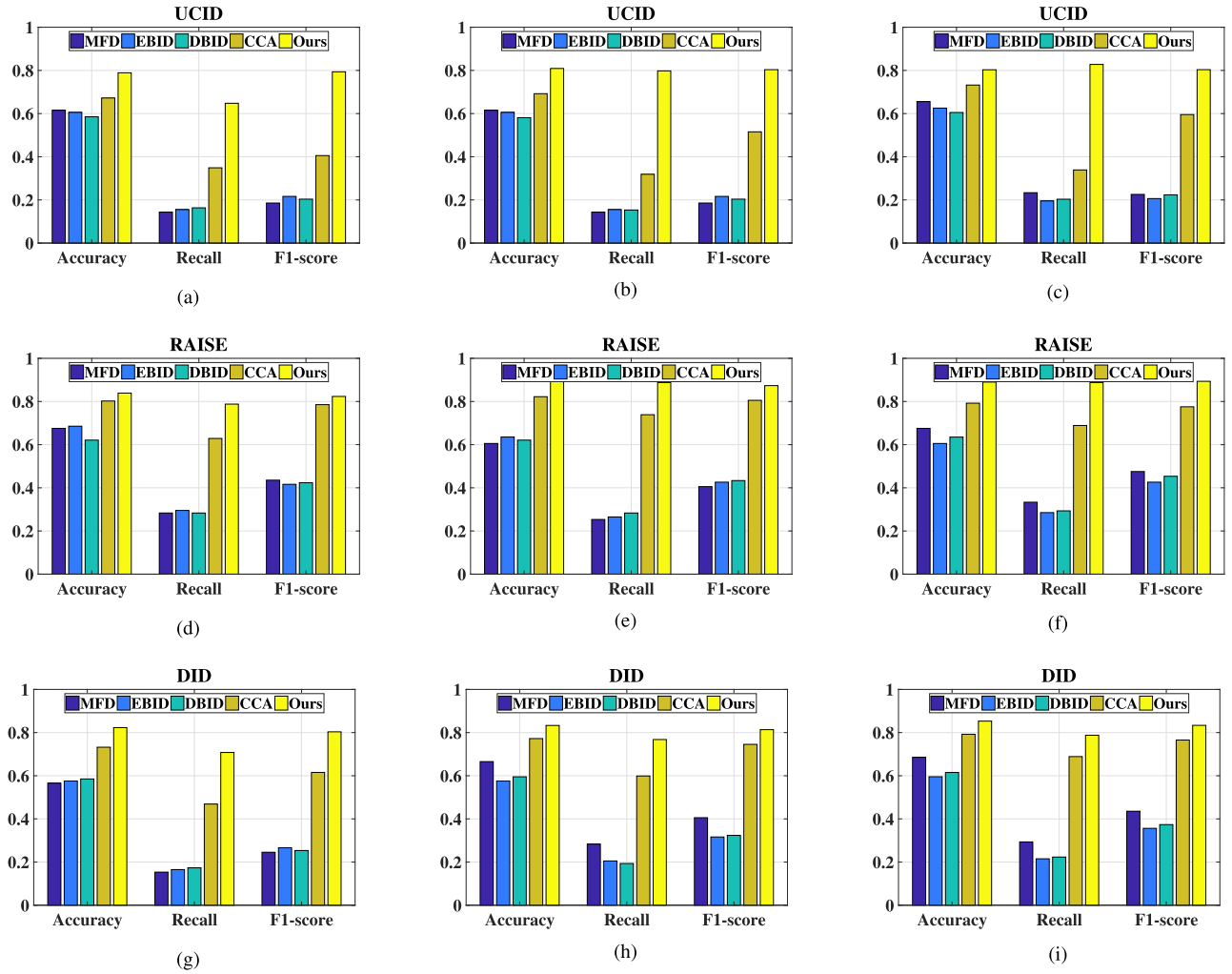


FIGURE 7. Performance comparisons of different methods on three datasets. (a) Block size = 32 × 32. (b) Block size = 64 × 64. (c) Block size = 128 × 128. (d) Block size = 32 × 32. (e) Block size = 64 × 64. (f) Block size = 128 × 128. (g) Block size = 32 × 32. (h) Block size = 64 × 64. (i) Block size = 128 × 128.

TABLE 1. Accuracy comparison with different δ values on the proposed framework.

Dataset	Predefined balance parameter δ					
	1	10	100	200	300	400
UCID	69.2%	66.7%	72.7%	**78.9%**	70.3%	68.6%
RAISE	59.0%	63.4%	82.6%	**83.8%**	78.6%	73.3%
DID	54.5%	58.1%	74.6%	**81.5%**	75.2%	70.7%

setting, 80% of the database is used for training, and the rest for testing.

We compared our method with four related methods, and these methods can be classified into three groups: 1) traditional image forensics: median filtering detection (MFD) [48]; 2) state-of-the-art detection for other types of inpainting: exemplar-based inpainting detection (EBID) [24], diffusion-based inpainting detection (DBID) [28]; and 3) the traditional CCA. All of the above 4 forensic methods achieve state-of-the-art performances for their own specialized tasks. The default parameters in these methods are used here.

Since inpainting creates blur areas, it is supposed that median filtering detection can distinguish them to some extent. However, traditional image forensics might fail for detecting sparsity-based inpainting because their tampering processes are obviously different. For the same reason, exemplar-based inpainting detection methods are also not ideal in the case of sparsity-based inpainting detection.

Li *et al.* [28] proposed a comprehensive study on diffusion-based inpainting detection and achieved promising performance for this type of inpainting detection. The feature extraction mainly depends on the changes in the image Laplacians along the direction of isophote. Since diffusion-based inpainting aims to preserve the isophote direction, their proposed feature set can properly characterize the traces of this type of image inpainting.

However, the inpainting principles and the traces left in inpainted images can be completely different between different types of inpainting methods. The feature set in [28] is not discriminative enough for sparsity-based inpainting, since

TABLE 2. Detection Accuracy Results for JPEG Compressed Inpainting Images.

Dataset	Method	32x32			64x64			128x128		
		QF = 90	QF = 70	QF = 50	QF = 90	QF = 70	QF = 50	QF = 90	QF = 70	QF = 50
UCID	MFD [48]	60.26%	57.53%	55.34%	60.08%	58.23%	55.56%	62.28%	53.36%	52.20%
	EBID [24]	56.44%	55.25%	53.67%	57.34%	55.78%	52.34%	58.55%	56.66%	53.60%
	DBID [28]	55.98%	55.38%	50.32%	57.21%	55.87%	52.55%	57.33%	54.32%	53.13%
	CCA	65.19%	62.75%	52.25%	68.20%	65.33%	55.20%	69.33%	63.24%	52.11%
	Ours	**70.98%**	**68.36%**	**59.23%**	**73.11%**	**68.80%**	**60.38%**	**75.36%**	**70.66%**	**63.28%**
RAISE	MFD [48]	61.37%	53.24%	55.25%	62.20%	55.33%	52.20%	63.33%	58.75%	52.36%
	EBID [24]	58.24%	55.36%	50.23%	60.16%	54.32%	51.64%	66.29%	56.68%	53.25%
	DBID [28]	60.35%	55.66%	50.60%	61.11%	55.80%	52.38%	60.55%	56.26%	53.91%
	CCA	66.45%	60.33%	52.78%	68.66%	61.89%	55.23%	72.62%	62.36%	55.85%
	Ours	**73.25%**	**70.39%**	**62.66%**	**75.70%**	**73.65%**	**63.97%**	**80.05%**	**75.50%**	**65.33%**
DID	MFD [48]	55.19%	52.88%	51.23%	62.59%	62.33%	55.20%	63.33%	53.24%	52.05%
	EBID [24]	56.98%	55.36%	50.66%	56.09%	55.29%	51.38%	59.55%	56.66%	53.11%
	DBID [28]	58.55%	55.32%	50.67%	59.11%	55.56%	52.39%	60.52%	53.22%	50.67%
	CCA	68.28%	57.55%	50.89%	72.67%	62.45%	55.88%	73.98%	69.56%	56.33%
	Ours	**72.95%**	**65.28%**	**60.20%**	**74.54%**	**68.65%**	**65.66%**	**79.15%**	**73.86%**	**63.56%**

the sparsity-based inpainting process is unrelated with the isophote direction. Fig. 7 shows the performance comparison of different methods. We can note that the diffusion-based inpainting detection method is not satisfactory for detecting the sparsity-based image inpainting.

We also compare the proposed method with traditional CCA. As mentioned before, in Section II-C and Section IV-B.2, sparsity-based inpainting operation strongly changes the correlations of data. By adding nuclear norm as a new regularization item, and selecting an appropriate balance parameter δ, we can get a more discriminative feature set as shown in Fig. 5. In the Fig. 7, the proposed method has better performance among all the datasets.

Although there is no existing baseline method in the current literature for sparsity-based inpainting detection, we can achieve competitive performances compared with other types of inpainting detection. More specifically, according to experimental results in [28], the F1-score obtained on the UCID dataset for diffusion-based inpainting detection is around 0.8, when the tampered regions are 32×32 squares. We obtain the same level F1-score for sparsity-based inpainting detection when the suspicious areas are 32×32 blocks for the UCID dataset.

D. ROBUSTNESS ANALYSIS

In this subsection, we elaborate the experimental results under the attacks of JPEG compression and Gaussian noise. These two common post operations are applied to inpainted images, respectively. The detection accuracy is used to demonstrate the effectiveness of our method and other four methods mentioned above.

1) ROBUSTNESS TO JPEG COMPRESSION

In practical applications, images are usually stored in JPEG format. It is critical to test the robustness of our method for JPEG compression. In this experiment, we investigate the performance of the proposed method under various compression conditions. The same image blocks in Section IV-C are used in this subsection. The original and inpainted image blocks are compressed with diverse quality factors (QF). Finally, the proposed detection algorithm is applied to the compressed image blocks and the accuracy results are reported in Table 2.

From this table, two main observations can be noted. First, as the image quality factor decreases, the detection accuracy reduces. When Q equals to 90, our detection accuracy only slightly degrades compared with uncompressed image blocks. The results support the robustness of the proposed algorithm for JPEG compression. Second, similar to the results in the previous section, our method is more effective for high-resolution images. The accuracy of RAISE is higher than that of UCID and DID.

2) ROBUSTNESS TO GAUSSIAN NOISE

To analyze the effects of noise, Gaussian noise is added to the images after inpainting. Images with different signal-to-noise ratios (SNRs) are obtained for testing. Generally speaking, noise is perceptually invisible when the SNR is greater than 45 dB. In most cases, image quality is considered acceptable when the SNR is between 36 to 45 dB, while image quality is considered to be poor for SNR below 36 dB [49]. The same image blocks in the previous subsection are used. The Gaussian noise is added to those original and inpainted image blocks, where SNR = 50, 40 and 30. We conducted our detection algorithm to image blocks with different SNRs. Table 3 shows the results.

As the SNR decreases, the performance degenerates. When SNR equals to 50, there is no significant performance degradation in our results. While when the SNR becomes 30, there is a clear accuracy degradation when compared with images from noiseless scenarios.

E. PERFORMANCE ON START-OF-THE-ART INPAINTED IMAGES

Image reconstruction can be viewed as an application of image inpainting. To demonstrate the generality of our detection algorithm, we also detect the inpainted images generated by Liu *et al.* [50], which is a state-of-the-art sparsity-based

TABLE 3. Detection Accuracy Results for Inpainting Images with Gaussian Noise.

Dataset	Method	32x32			64x64			128x128		
		SNR=50	SNR=40	SNR=30	SNR=50	SNR=40	SNR=30	SNR=50	SNR=40	SNR=30
UCID	MFD [48]	61.22%	58.75%	55.34%	61.20%	57.36%	55.20%	63.33%	53.24%	52.11%
	EBID [24]	59.25%	55.36%	50.23%	59.26%	54.80%	52.39%	61.53%	56.66%	53.50%
	DBID [28]	57.97%	55.36%	50.50%	59.11%	55.59%	52.35%	60.88%	56.50%	52.11%
	CCA	65.53%	63.96%	59.40%	67.32%	60.56%	55.28%	69.60%	54.35%	53.29%
	Ours	**73.66%**	**64.86%**	**62.13%**	**73.20%**	**66.33%**	**64.89%**	**74.67%**	**68.23%**	**65.21%**
RAISE	MFD [48]	59.19%	56.55%	53.25%	62.26%	58.33%	56.56%	64.33%	53.90%	52.55%
	EBID [24]	58.50%	53.28%	50.95%	60.11%	55.80%	53.38%	61.55%	56.39%	53.33%
	DBID [28]	57.16%	56.32%	50.16%	59.66%	55.56%	54.55%	60.64%	55.66%	53.20%
	CCA	66.26%	59.58%	56.35%	67.53%	59.10%	52.95%	67.53%	53.52%	52.55%
	Ours	**74.33%**	**63.22%**	**60.65%**	**75.10%**	**69.36%**	**65.33%**	**76.66%**	**69.36%**	**65.30%**
DID	MFD [48]	56.10%	55.75%	52.85%	56.99%	55.33%	53.55%	59.65%	53.80%	52.36%
	EBID [24]	56.98%	55.86%	51.50%	57.11%	56.20%	53.50%	60.54%	57.25%	53.25%
	DBID [28]	55.20%	54.36%	50.23%	56.53%	54.88%	52.38%	60.75%	56.38%	53.11%
	CCA	61.36%	59.57%	53.28%	63.66%	61.17%	55.97%	64.25%	62.50%	58.85%
	Ours	**69.51%**	**64.83%**	**62.25%**	**68.95%**	**65.62%**	**63.30%**	**73.12%**	**67.39%**	**62.88%**

TABLE 4. Performance Comparisons for the State-of-the-art Sparsity-based Image Reconstruction.

Method	Accuracy	Recall	F1-score
MFD [48]	0.5322	0.1925	0.3024
EBID [24]	0.5269	0.1785	0.2580
DBID [28]	0.5534	0.2069	0.3278
CCA	0.5628	0.2265	0.3406
Ours	**0.6736**	**0.3526**	**0.4128**

method for image restoration. The author provides 10 original images and 10 reconstructed images with size 512 × 512. We cut them into blocks with size 32 × 32. We therefore have 5,120 blocks in total, including 2,560 original blocks and 2,560 reconstructed blocks. The experiments are conducted using the same process as in the previous section. The results are reported in Table 4.

Liu *et al.* utilized more advanced techniques for sparsity-based image reconstruction. It would be more difficult to justify whether the images have been modified. As the experimental results show, the detection performances are lower than that in Elad's work. However we still obtain better performances compared with other traditional image forensic methods [48] and other types of image inpainting detection methods [24], [28].

Ideally we would like to intensively test the proposed detection method on the state-of-the-art sparsity-based method in [50]. However we can not get the codes to reproduce all required images and we only managed to obtain limited example images from the authors for evaluation.

V. CONCLUSION

In this paper, we proposed an image forensic method to detect sparsity-based image inpainting, a useful image editing operation with a broad range of applications. Completely different from other types of image inpainting, sparsity-based inpainting has a potential connection with CCA. To enhance the inter-class difference in our feature set, a novel objective function is proposed. We also developed a corresponding optimization algorithm to solve the new function. Experimental results showed better performance of the proposed method when compared with other detection methods.

In the future, we plan to further improve the detection performance of sparsity-based image inpainting by integrating other techniques. For example, subspace learning can be used to project the tampered signals into different subspaces. It would also be beneficial to seek a more flexible mechanism and a more general method to detect different types of image inpainting.

REFERENCES

[1] H. Farid, "Image forgery detection," *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 16–25, Mar. 2009.
[2] J. Fridrich, "Digital image forensics," *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 26–37, Mar. 2009.
[3] A. Swaminathan, M. Wu, and K. J. R. Liu, "Component forensics," *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 38–48, Mar. 2009.
[4] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758–767, Feb. 2005.
[5] Y. Su, X. Jin, C. Zhang, and Y. Chen, "Hierarchical image resampling detection based on blind deconvolution," *J. Vis. Commun. Image Represent.*, vol. 48, pp. 480–490, Oct. 2017.
[6] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 507–518, Mar. 2015.
[7] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 6, pp. 1841–1854, Dec. 2012.
[8] J. Lukáš, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 205–214, Jun. 2006.
[9] M. Chen, J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.
[10] W. Luo, Y. Wang, and J. Huang, "Detection of quantization artifacts and its applications to transform encoder identification," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 810–815, Dec. 2010.
[11] T. Bianchi and A. Piva, "Detection of nonaligned double JPEG compression based on integer periodicity maps," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 842–848, Apr. 2012.
[12] H.-D. Yuan, "Blind forensics of median filtering in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1335–1345, Dec. 2011.
[13] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu, "Robust median filtering forensics using an autoregressive model," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 9, pp. 1456–1468, Sep. 2013.
[14] G. Cao, Y. Zhao, R. Ni, and X. Li, "Contrast enhancement-based forensics in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 3, pp. 515–525, Mar. 2014.

[15] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 492–506, Sep. 2010.

[16] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6882–6890.

[17] X. Li, Y. Ye, and X. Xu, "Low-rank tensor completion with total variation for visual data inpainting," in *Proc. AAAI*, San Francisco, CA, USA, 2017, pp. 2210–2216.

[18] F. Voci, S. Eiho, N. Sugimoto, and H. Sekibuchi, "Estimating the gradient in the Perona-Malik equation," *IEEE Signal Process. Mag.*, vol. 21, no. 3, pp. 39–65, May 2004.

[19] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[20] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer-Verlag, 2010.

[21] Q. Wu, S.-J. Sun, W. Zhu, G.-H. Li, and D. Tu, "Detection of digital doctoring in exemplar-based inpainted images," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 3, Jul. 2008, pp. 1222–1226.

[22] K. S. Bacchuwar, A. Deep, and K. R. Ramakrishnan, "A jump patch-block match algorithm for multiple forgery detection," in *Proc. Int. Multi-Conf. Autom., Comput., Commun., Control Compress. Sens. (iMac4s)*, Mar. 2013, pp. 723–728.

[23] I.-C. Chang, J. C. Yu, and C.-C. Chang, "A forgery detection algorithm for exemplar-based inpainting images using multi-region relation," *Image Vis. Comput.*, vol. 31, no. 1, pp. 57–71, 2013.

[24] Z. Liang, G. Yang, X. Ding, and L. Li, "An efficient forgery detection algorithm for object removal by exemplar-based image inpainting," *J. Vis. Commun. Image Represent.*, vol. 30, pp. 75–85, Jul. 2015.

[25] D. Zhang, Z. Liang, G. Yang, Q. Li, L. Li, and X. Sun, "A robust forgery detection algorithm for object removal by exemplar-based image inpainting," *Multimedia Tools Appl.*, vol. 77, no. 10, pp. 11823–11842, May 2018.

[26] X. H. Li, Y. Q. Zhao, M. Liao, F. Y. Shih, and Y. Q. Shi, "Detection of tampered region for JPEG images by using mode-based first digit features," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, p. 190, 2012.

[27] Y. Q. Zhao, M. Liao, F. Y. Shih, and Y. Q. Shi, "Tampered region detection of inpainting JPEG images," *Optik-Int. J. Light Electron Opt.*, vol. 124, no. 16, pp. 2487–2492, 2013.

[28] H. Li, W. Luo, and J. Huang, "Localization of diffusion-based inpainting in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 3050–3064, Dec. 2017.

[29] M. Ghorai, S. Mandal, and B. Chanda, "A group-based image inpainting using patch refinement in MRF framework," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 556–567, Feb. 2018.

[30] J. Ho Lee, I. Choi, and M. H. Kim, "Laplacian patch-based image synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2727–2735.

[31] Z. Li, H. He, H. M. Tai, Z. Yin, and F. Chen, "Color-direction patch-sparsity-based image inpainting using multidirection features," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 1138–1152, Mar. 2015.

[32] P. Buyssens, M. Daisy, D. Tschumperlé, and O. Lézoray, "Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1809–1824, Jun. 2015.

[33] T. Ogawa and M. Haseyama, "Adaptive subspace-based inverse projections via division into multiple sub-problems for missing image data restoration," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5971–5986, Dec. 2016.

[34] V. Kumar, J. Mukherjee, and S. K. D. Mandal, "Image inpainting through metric labeling via guided patch mixing," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5212–5226, Nov. 2016.

[35] Q. Guo, S. Gao, X. Zhang, Y. Yin, and C. Zhang, "Patch-based image inpainting via two-stage low rank approximation," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 6, pp. 2023–2036, Jun. 2018.

[36] C. Guillemot and O. Le Meur, "Image inpainting: Overview and recent advances," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 127–144, Jan. 2014.

[37] G. Andrew, R. Arora, K. Livescu, and J. Bilmes, "Deep canonical correlation analysis," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Atlanta, GA, USA, 2013, pp. 1247–1255

[38] K. B. Petersen *et al.*, "The matrix cookbook," *Tech. Univ. Denmark*, vol. 7, no. 15, p. 510, 2008.

[39] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, "Fast and accurate matrix completion via truncated nuclear norm regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2117–2130, Sep. 2013.

[40] J. Li, Y. Wu, J. Zhao, and K. Lu, "Low-rank discriminant embedding for multiview learning," *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3516–3529, Nov. 2017.

[41] P. Li, J. Yu, M. Wang, L. Zhang, D. Cai, and X. Li, "Constrained low-rank learning using least squares-based regularization," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4250–4262, Dec. 2017.

[42] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[43] J.-F. Cai, E. J. Candes, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.

[44] G. Schaefer and M. Stich, "UCID: An uncompressed color image database," *Proc. SPIE*, vol. 5307, pp. 472–481, Dec. 2003.

[45] D.-T. Dang-Nguyen, C. Pasquini, V. Conotter, and G. Boato, "RAISE: A raw images dataset for digital image forensics," in *Proc. ACM Multimedia Syst. Conf.*, New York, NY, USA, 2015, pp. 219–224.

[46] T. Gloe and R. Böhme, "The 'dresden image database' for benchmarking digital image forensics," in *Proc. ACM Symp. Appl. Comput.*, New York, NY, USA, 2010, pp. 1584–1590.

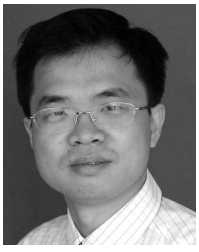
[47] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, Art. no. 27.

[48] J. Chen, X. Kang, Y. Liu, and Z. J. Wang, "Median filtering forensics based on convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 1849–1853, Nov. 2015.

[49] X. Feng, I. J. Cox, and G. Doerr, "Normalized energy density-based forensic detection of resampled images," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 536–545, Jun. 2012.

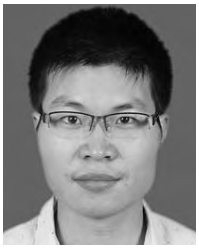
[50] X. Liu, D. Zhai, J. Zhou, S. Wang, D. Zhao, and H. Gao, "Sparsity-based image error concealment via adaptive dual dictionary learning and regularization," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 782–796, Feb. 2017.

XIAO JIN received the B.S. degree in electronic information engineering from Tianjin University, Tianjin, China, where he is currently pursuing the Ph.D. degree with the School of Electrical and Information Engineering. His current research interests include multimedia content analysis and security.



YUTING SU received the M.S. and Ph.D. degrees in electrical engineering from Tianjin University, Tianjin, China, in 1995, 1998, and 2001 respectively. He is currently a Professor with the School of Electrical and Information Engineering, Tianjin University. He was a Visiting Scholar with Case Western Reserve University, Cleveland, OH, USA, from 2009 to 2010. His research interests include multimedia content analysis and security.

LIANG ZOU received the M.Sc. degree in biomedical engineering from the University of Science and Technology of China in 2013 and the Ph.D. degree in electrical and computer engineering from The University of British Columbia, Canada, in 2017. He is currently a Post-Doctoral Research Fellow with the School of Computer Science and Engineering and the Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly, Nanyang Technological University, Singapore. His current research interest includes statistical signal processing, machine learning, and medical imaging.



YONGWEI WANG received the B.S. and M.S. degree from Northwestern Polytechnical University, China, in electronic information engineering. He is currently pursuing the Ph.D. degree with the Department Electrical and Computer Engineering, The University of British Columbia, Canada. His current research interests include multimedia content analysis and security.

PEIGUANG JING received the M.S. and Ph.D. degrees in signal and information processing from Tianjin University, Tianjin, China, in 2012 and 2017, respectively. He is currently an Assistant Professor with the School of Electrical and Information Engineering, Tianjin University. He was a Visiting Student with the NExT Center, National University of Singapore, from 2014 to 2015. His current research interests include multimedia content analysis and computer vision.

Z. JANE WANG received the B.S. degree from Tsinghua University, China, in 1996, and the M.S. and Ph.D. degrees from the University of Connecticut in 2000 and 2002, respectively, all in electrical engineering. She was a Research Associate with the Electrical and Computer Engineering Department, University of Maryland at College Park, College Park. Since 2004, she has been with the Department Electrical and Computer Engineering, The University of British Columbia, Canada, where she is currently a Full Professor. She has authored over 110 journal papers and 100 peer-reviewed conference papers. Her research interests include statistical signal processing, machine learning, and medical imaging. She has been an organizing committee member for numerous IEEE conferences and workshops and an associate editor for several IEEE journals.