

# Inpainting: Survey and Experiments

Chang Shu and Yaojie Liu

*University of Electronic Science and Technology of China, Chengdu, China*  
*The Ohio State University, Columbus, USA*  
*changshu@uestc.edu.cn, liu.4002@osu.edu*

## Abstract

*Image inpainting is an important research topic in the field of image processing. The goal of inpainting is to recover the lost information of the target region from the rest of the image. Inpainting techniques can be applied in areas such as old photo restoration, object removal and demosaicing. Based on previous literatures of image inpainting and image modeling, we categorize inpainting processes of different methods and algorithms into the structure layer and the texture layer. Then the mathematical inpainting models and the formations of image impairment are analyzed and evaluated in detail. Experimental results are provided in the fifth section regarding to eight different algorithms measured by Peak Signal to Noise Ratio (PSNR) as well as direct visual perception.*

**Keywords:** *image inpainting, variational method, partial differential equation (PDE), exemplar based method, sparse representation*

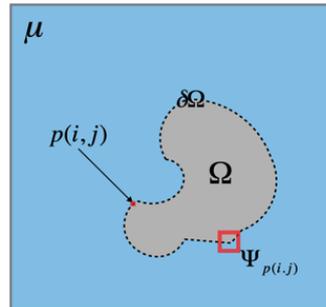
## 1. Introduction

Image inpainting is the technique to modify and restore damaged images in a visually plausible way. In retrospect, this technique can be dated back to the Renaissance when medieval artwork artisans retouched the faded area in those deteriorated paintings using the paintbrush. According to this ancient artistry, in 2000, Bertalmio et al. initially proposed an inpainting algorithm in [1]. The underlying philosophy of this inpainting algorithm is simple and intuitive: it resembles the way how human artisans repair an incomplete picture, to create smooth transition at the border and uniform color fluid and texture inside the unknown area. Inpainting algorithms compute and understand the existing part of the image (*i.e.*, the source region in Bertalmio's terminology), and mathematically infer what we believe the pattern is in the missing holes (*i.e.*, the target region). In a broad sense, image inpainting techniques follow one macro mechanism: 'visually acceptability'.

While image inpainting is regarded as a branch of image restoration, image inpainting and the traditional restoration problems are different. In traditional problems such as haze removal and motion deblurring, target region is damaged but not totally unknown. On the other hand, in the inpainting issues, information can only be inferred from the outside of the target region. Image inpainting has been developed through out the past two decades, early examples include: scratch removal of digital photos and old films [1-2], text erasing [1-4]. Current inpainting techniques are applicable widely in special effects such as object removal [1-6], zooming, demosaicing and super-resolution [2-7], image coding and decoding [2], and 3D reconstruction [8] as well.

As Shen talked in [9], image is generated through the simulation of the underlying physical, chemical, and biological processes. Generally all types of image degradation can be classified into these three categories. Dealing with various types of image degradation, we argue that finding out a proper and valid image model could greatly improve the inpainting performance. Representative models include Bertalmio-Sapiro-Caselles-Ballester (BSCB) model [1], Mumford-Shah model [10-11], Navier-

Stokes model [12], copy-and-paste patterns simulated by iterated function systems [13] and *etc.* In this paper, we consider common image models listed above and divide the inpainting process into the structure layer and the texture layer for further discussion. Terminologies used in this paper are shown in Figure 1.



**Figure 1. Inpainting Model Notations:  $\mu$  Denotes a Complete Image Domain, often a Rectangular Area. Target Region is  $\Omega$  and the other Part of the Image is the Source Region  $\Omega^c$  (and  $\Omega^c \cup \Omega = \mu$ ). The Outline of Target Region  $\Omega$  is Amorphous and is Signified as  $\delta\Omega$ . Current Inpainting Pixel is Represented by  $p(i,j)$  and the Inpainting Patch Centered at  $p(i,j)$  is Symbolized as  $\Psi_{p(i,j)}$**

The rest of this paper is arranged as follows: in Section 2 and Section 3 we introduce and compare various algorithms according to their performance on structure and texture respectively. In Section 4, inpainting based on sparse representation is described with respect to both structure and texture restoration. In Section 5, the results of the generic evaluation of texture and structure are provided. Experiments are carried out on Matlab. The test data consists of synthesized picture and nature photos, and the artificial missing regions are made on crucial areas of the original images. Conclusion are given in Section 6.

## 2. Structure

Structure contains two elements: discontinuities and smooth variation. Discontinuities often exist within the edges and outline of an object and smooth variation is the overall grey scale change due to light, pose, and *etc.* In [14], Shen listed three principles of structure reconstruction as:

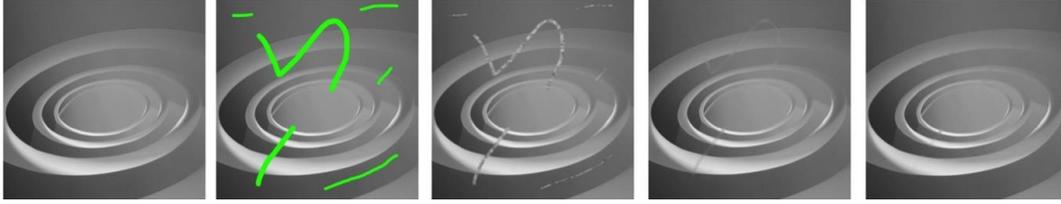
- I. Locality: inpainting information comes from the neighboring pixels;
- II. Reasonable edges restoration;
- III. Robustness on noise.

### 2.1. Partial Differential Equations Methods

Partial Differential Equations (PDE) are the foundation of the diffusion inpainting methods (also called variational methods). Before inpainting the image, user only needs to select the target region (the green area shown in Figure 2. PDE methods propagate the structure from boundary  $\delta\Omega$  inwards, using differential equation (mainly partial differential, in discrete and iterative way), which can be written as

$$I^{n+1}(i,j) = I^n(i,j) + \Delta t I_t^n(i,j), \forall (i,j) \in \Omega \quad (1)$$

where the super-index  $n$  demotes the iterative time,  $I(i, j)$  is the pixel at  $(i, j)$  in the target area  $\Omega$ ,  $\Delta t$  is the step size of improvement and  $I_t^n(i, j)$  denotes the inpainting variation (diffusion value) of each iteration.



**Figure 2. Result of the Processing of Variational Algorithm (BSCB's Method) for a Scratch Recovery**

In [1], Bertalmio proposed an inpainting method based on PDE that prolongs the isophote lines to propagate the structure into the gap. Specifically the inpainting variation is composed of an isophote direction term  $\vec{N}$  (the vector perpendicular to isophote line) and the differential of the image smoothness estimator  $\vec{L}$ , that is, term  $\delta\vec{L}$  in (2). To avoid isophote lines crossing each other, an anisotropic diffusion (3) is added B steps every A steps of inpainting.

$$I_t^n(i, j) = \delta\vec{L}(i, j) \cdot \vec{N}^n(i, j) \quad (2)$$

$$\frac{\partial I}{\partial t}(i, j, t) = \text{smooth}_\varepsilon(i, j) \kappa(i, j, t) |\nabla I(i, j, t)|, \forall (i, j) \in \Omega^\varepsilon \quad (3)$$

where  $\Omega^\varepsilon$  is the area  $\varepsilon$  wider than  $\Omega$ ,  $\text{smooth}_\varepsilon$  is a smooth function on  $\Omega^\varepsilon$ , and  $\kappa(i, j, t)$  is the Euclidean curvature of the isophotes of  $I$ . Later Bertalmio proposed a method based on Navier-Stokes equations, treating the isophote as fluid [12]. Bertalmio's algorithm can restore smooth color fluid, but it would blur the edge and discontinuities.

Total Variational (TV) proposed in [2], process variational inpainting in a different way. Inpainting variation is computed as (4). Instead of using partial differential, TV method process central differential of neighboring half-pixel values shown in (4).

$$I_t^n(i, j) = \nabla j^{(n)} = \frac{j_{(\frac{1}{2}, 0)}^1 - j_{(-\frac{1}{2}, 0)}^1}{h} + \frac{j_{(0, \frac{1}{2})}^2 - j_{(0, -\frac{1}{2})}^2}{h} \quad (4)$$

where  $h$  denotes the grid size (often set as 1). In (4),  $j$  can be represented as  $-\nabla I / |\nabla I|$ , so the diffusion intensity of TV is  $1/|\nabla I|$ . Therefore TV doesn't rely on the geometry information of isophotes. Compared with Bertalmio's method, TV algorithm can keep the repaired edge sharp and clean. However, Chan argued that good inpainting mechanism should follow connectivity principle of vision psychology in [15-16]. One drawback is that TV algorithm cuts the continuous object apart, which somehow would potentially cause artefacts.

Later in [4], curvature-driven diffusion algorithm (CDD) was proposed to overcome the disadvantage of TV in filling-in the whole object (especially the edges). In CDD method, the conductivity coefficient is modified to  $j = g(|\kappa|) \cdot \nabla I / |\nabla I|$ , where  $\kappa$  is a scalar curvature and function  $g(x)$  is for stabilization. While CDD method keep continuous object connected,  $g(|\kappa|)$  term highly increases the computing time of CDD. A fast inpainting approach was purposed in [17]. It convolutes the target area till convergence by

a constant matrix, both the computing and converge time are reduced compared with previous methods.

## 2.2. Exemplar Based Methods

Differently, exemplar-based methods use the copy-and-paste strategy to fill pixel block into the target region and calculates a proper filling order to ensure correct structure propagation, which is called Patch Selection step in exemplar-based methods (details about the copy-and-paste strategy can be found in the texture section). By computing a priority value, exemplar-based algorithms assign pixels on the continuation of image structures with higher priority to ensure them to be filled earlier. In Criminisi's method [5], the priority term (*i.e.*, inpainting order) consists of confidence term  $C(p)$  and data term  $D(p)$ . They represent the measure of the available information around the pixel  $p$  and the measure of isophotes strength and structure at point  $p$  respectively [5,6], as shown in (5), (6).  $D(p)$  contains information about isophote.  $C(p)$  and  $D(p)$  guarantee a well ordered propagation of the edges while avoiding indefinite growth of the edges at the same time. The priority term would be updated every time when a pixel block is filled. This algorithm iteratively processes until all target regions are filled.

$$C(p) = \frac{\sum_{q \in \Psi_p \cap \Phi} C(q)}{\Psi_p} \quad (5)$$

$$D(p) = \frac{|\nabla I_p^\perp \cdot \vec{n}_p|}{\alpha} \quad (6)$$

Many ways to describe the priority term were proposed afterwards as appeared in [18] and [19]. Other methods as in [20] select several images with different scales (multi-resolution) to inpaint several times to recover the structure. Additionally, the method proposed by Sun in [21] enhances the structure propagation capability of exemplar-based method, by means of manually specification of the line information of the missing area.

## 3. Texture

As mentioned in section 1, texture contains repetitive two-dimensional patterns with moderate stochasticity [6]. It can be modeled by Markov Random Fields (MRF) and Gibbs Fields (GF) [22-23]. Early work in [24-25] proposed effective algorithms to deal with pure texture. Recent algorithms employ the copy-and-paste techniques to synthesize texture, where the copy patch is chosen from the source region  $\Omega_c$ . One texture recovery example is shown in Figure 3.

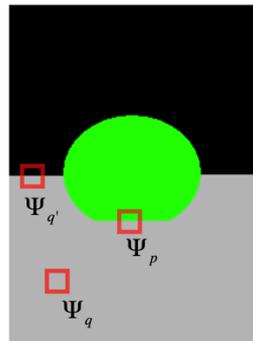


Figure 3. Example of Exemplar Based Inpainting

Basically, copy-and-paste strategy fills the missing region pixel by pixel from the boundary  $\delta\Omega$  inwards. Once the pixel  $p(i, j \in \delta\Omega)$  was chosen, pixel  $p(i, j)$  and its surroundings will be bound as a square patch  $\Psi_p$ . Since the pixel is on the boundary, patch  $\Psi_p$  would contain a potential texture pattern while some pixels remain unknown. Then the algorithm would fetch the best match  $\Psi_{\hat{q}}$  in the source region measured by certain criteria such as sum of squared differences (SSD) in [5], appearance-space in [26], correspondence maps in [27], and normalized cross correlation (NCC) in [28], between  $\Psi_p$  and  $\Psi_{\hat{q}}$ , shown in (7). This process iterates till all the missed pixels are filled-in.

$$\Psi_{\hat{q}} = \arg \min d(\Psi_p, \Psi_q) \quad (7)$$

One problem of the copy-and-paste strategy is that the search process is time-consuming. In [29], an enhanced adaptive search method was proposed, altering the searching window based on the structure sparsity of the selected patch. Also in order to reduce the computational complexity of the algorithm, a simplified method was adopted in [30], with the assumption that most often the patch  $\Psi_{\hat{q}}$  that most resembles the selected patch  $\Psi_p$  lies very close to the patch selected to be inpainted.



**Figure 4. Patches with Same Minimum SSD. For  $\Psi_p$ ,  $\Psi_q$  and  $\Psi_{q'}$ , Keep the Same Minimum SSD, but Obviously Selecting  $\Psi_{q'}$  is Incorrect and May Not Produce the Most Visually Plausible Result**

Another problem is that the distance (*e.g.*, SSD distance) of some unbecoming bad patch may be the same as a good patch, shown in Figure 4. In [30] pixels are randomly chosen to fill in from any matches  $\Psi_{\hat{q}_i}$  under a certain SSD threshold. And variance is computed as a second criterion in [31] when this case happened. In [32] the author proposed the weighted aggregation of nonlocal-means information of different  $\Psi_{\hat{q}}$ . To find a better distance measure is another way to address this problem.

In addition, optimizing coherence is another significant approach to improve texture inpainting performance. When two blocks close to each other are both filled by patches far from them, artefact often occurs among edges of filled patches (in patch-based method). Spatial patch blending mentioned in [33] is an effective post-processing to diminish such artefact, and a fast blending version was proposed in [34]. However, extra information such as filling map is needed. Differently, the method in [35] offers simultaneous blending processing. In [36],  $\Psi_q$  is modified to satisfy that the boundary of  $\Psi_q$  has the same image gradient (relative information) with the boundary of  $\Psi_p$  (where  $\Psi_q$  should be put in). In principle, Poisson-guided interpolation is introduced to solve a minimization problem by Poisson equation with Dirichlet boundary conditions.

## 4. Sparse Inpainting

### 4.1. Sparse Compression

In traditional signal processing, sparse representation has been proven to have an excellent performance in describing and compressing signals with high dimension. In recent research, sparse representation and sparse coding has been applied to image inpainting techniques. The essence of this approach is to represent image by sparse combination of an overcomplete set of transforms (*e.g.*, wavelet, contourlet, DCT, or a dictionary obtained by training), and then the target region is inferred by adaptively updating this sparse representation:

$$\underline{x} = D\underline{\alpha} \text{ s.t. } \|\underline{\alpha}\|_0 \leq L \quad (8)$$

where vector  $x$  denotes the original signal,  $\chi \in R^n$ ,  $D$  denotes the dictionary  $D \in R^{n \times k}$ , containing  $k$  columns, and each column is called an atom. For sparse representation,  $k > n$ , so it is a over-complete dictionary (or redundant dictionary). Vector  $\alpha$  is a sparse linear combination of atoms in  $D$ ,  $\|\alpha\|_0 \leq L$ . Frameworks of sparse representation and related mathematical models are proposed in [37] and [38].

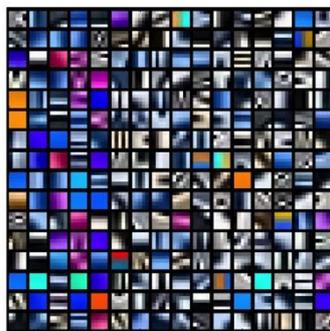
### 4.2. Computing the Sparse Representation Matrix

Suppose the dictionary and image vector are obtained, the computational problem is NP-hard (9). To solve it, there are relaxation methods such as Basis Pursuit (BP, also known as LASSO method [39]) to convert the NP-hard problem to L1 convex problem, and Greedy methods such as Matching Pursuit(MP), Orthogonal MP(OMP) [40-42], or hybrid methods such as StOMP [43], CoSaMP [44].

$$\min_{\underline{\alpha}} \|\underline{D}\underline{\alpha} - \underline{y}\|_2^2 \text{ subject to } \|\underline{\alpha}\|_0 \leq L \quad (9)$$

### 4.3. Dictionary

To construct a dictionary, we could use different transform, *e.g.*, wavelets of different bases, Curvelets, DCT, *etc.*, or directly train a dictionary from sample images. K-SVD algorithm provides an effective way to learn the dictionary [45]. For image inpainting, we often learn the dictionary in two ways. One is using a specific dataset of images. This is good for inpainting problems where the target area is very special, *e.g.*, the face. The second one is using the corrupted image itself. Generally, patch (*e.g.*, a 5\*5\*3 pixels block for RGB image) is selected to be sparse represented, shown in Figure 5. The patch size depends on the image size and the target region size.



**Figure 5. An Example of Sparse Representation Dictionary. Each Block Represents an Atom of 5\*5\*3**

Figure is taken from [49].

#### 4.4. Sparse-Representation-Based Inpainting

Sparse representation was introduced into image inpainting in 2005 as morphological component analysis (MCA) in [38]. MCA method decomposes the image into texture layer and structure layer and uses two incoherent adapted dictionaries to recovery the two layers separately. While inpainting each part, MCA solve the minimization problem with  $l_0$ -norm term with total-variational (TV) as constraints:

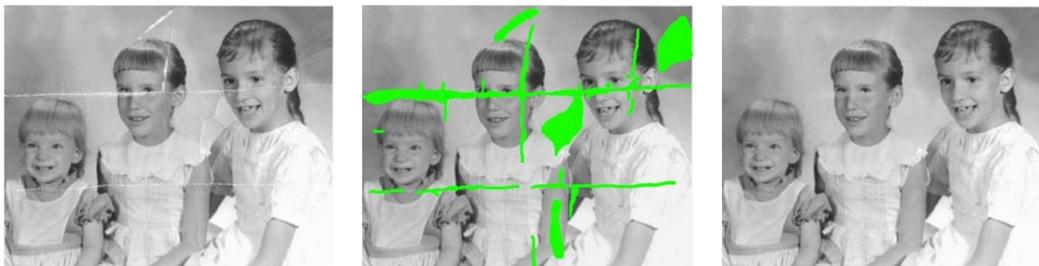
$$\{\underline{\alpha}_t^{opt}, \underline{\alpha}_n^{opt}\} = \arg \min_{\{\underline{\alpha}_t, \underline{\alpha}_n\}} \|\underline{\alpha}_t\|_1 + \|\underline{\alpha}_n\|_1 + \lambda \|X - T_t \underline{\alpha}_t - T_n \underline{\alpha}_n\|_2^2 + \gamma TV\{T_n \underline{\alpha}_n\}. \quad (2)$$

where  $l_1$ -norm is chosen to replace  $l_0$ -norm,  $T_t$  is the dictionary for texture and  $T_n$  is for structure. Off-the-shelf dictionary is proposed in [46-47], sparse representation is combined with exemplar-based method, computing the fill-in order first and recovering the patch by sparse representation iteratively. Patch sparse representation applies well in [47-48], which combined the sparse representation with the exemplar based texture synthesis.

Likewise, Xu proposed a patch sparsity method in the framework of sparse representation and exemplar-based method, replacing the traditional isophote-based priority with structure sparsity priority. Priority term  $P(p)$  is represented by structure sparsity among the patch and its surroundings [48]. Unlike previous methods, Xu don't construct a dictionary. Instead the algorithm chooses the top-N matched patches to inpaint the selected pixel and optimizes the combination coefficients of those candidates to ensure the sparseness of linear combination. This method shows a great robustness without much computation time sacrifice. More valuable work of sparse representation in the application of image inpainting are in [49-50].

### 5. Experiment

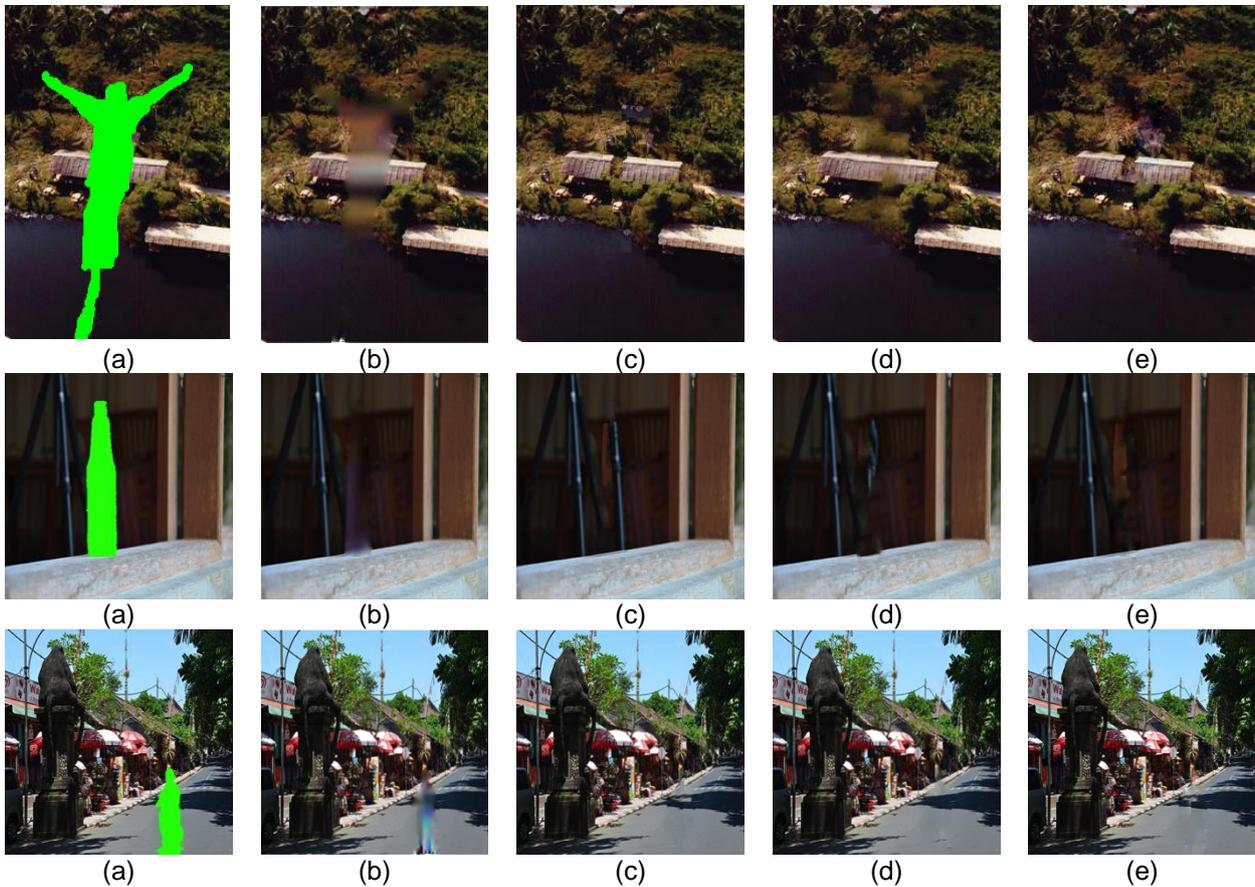
We implement experiments and compare the results in both visual perspective and quantitative perspective. Based on previous work in [51], we enhance the comprehensiveness of our experiment. Experiment images consist of both natural and synthetic images and target regions are created artificially on plenty of areas including the area that contains complex structure and texture information. The performance of algorithms is measured by peak signal-to-noise ratio (PSNR), where signal information comes from the original image (we assumed original image is the only solution of the test image). Additionally, we evaluate performance of different algorithms based on individuals' visual perceptions and feelings.



**Figure 6. One Example of Scratch and Crease Recovery**

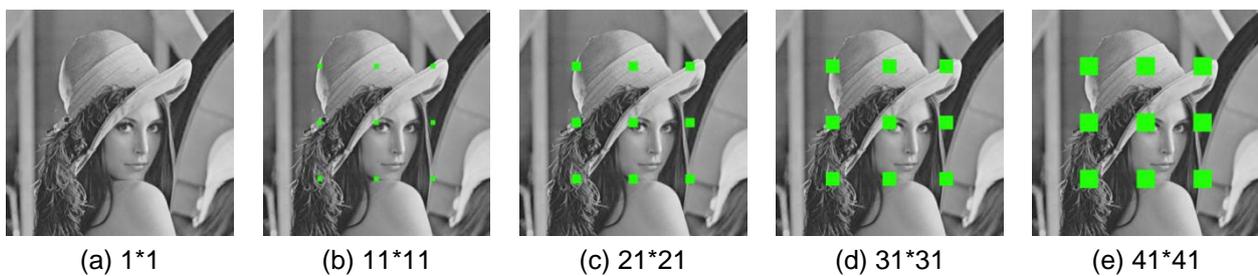
One inpainting example is presented in Figure 6. As shown in the figure, target region is often large and amorphous. Moreover, target region would cover special image information such as the left eye of the second girl in Figure 6, and thus inpainting artefact (e.g., inappropriate recovering structure and texture) would easily generate. We evaluate the

performances of 8 algorithms (*i.e.*, BSCB method [1], TV method [2], CDD method [4], Fasting inpainting [17], Criminisi method [6], Non-local mean method [32], and Poisson method [35]) in various ways. Firstly, we choose images with undesired objects to be removed to test the above methods. And the results are shown in Figure 7.

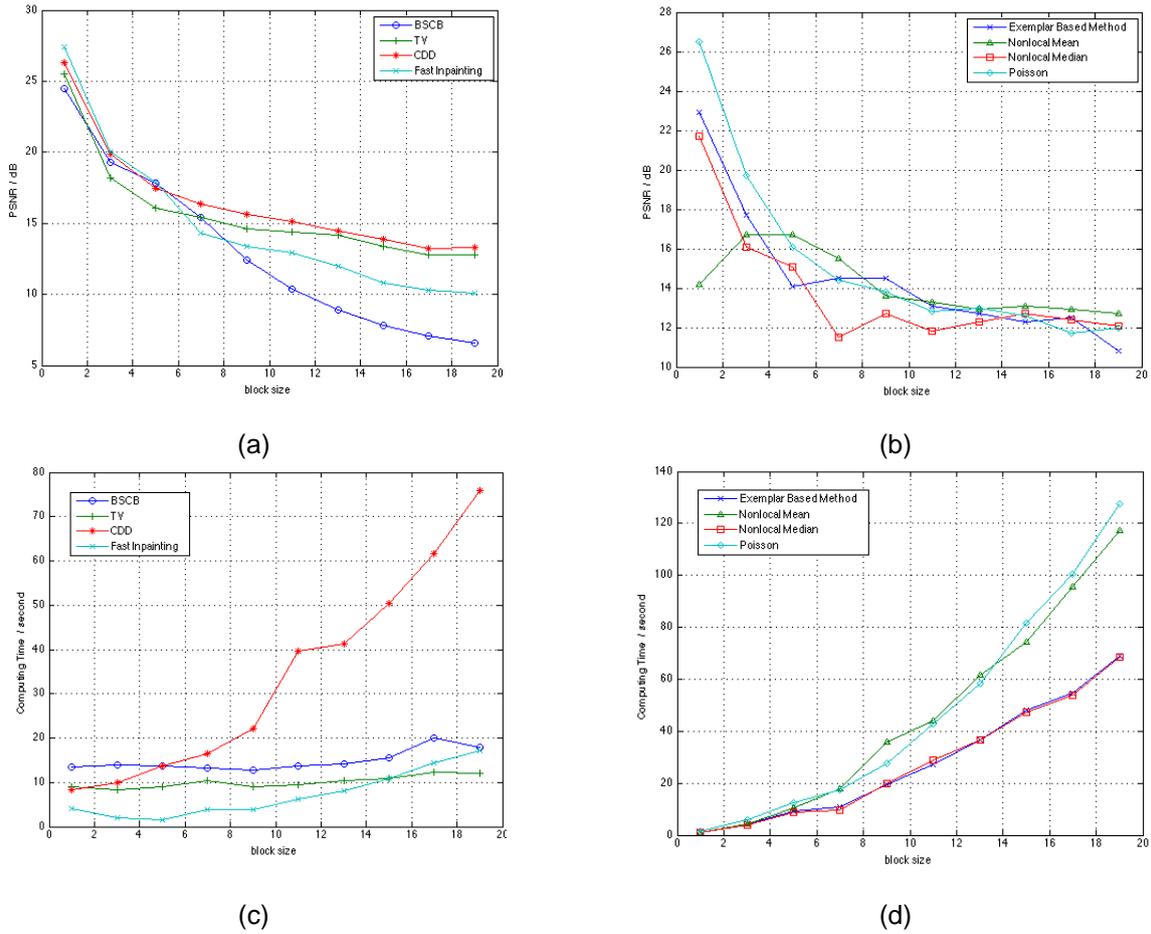


**Figure 7. Results of Object Removal. (a) Masked Image; (b) BSCB Method; (c) Criminisi Method; (d) Non-Local Mean Method; (e) Poisson Method**

From Figure 7, we found that variational methods such as BSCB method have a worse performance when target region is relatively large (mostly in situations of objects removal) since they cause a fact of smoothing. In addition, we want to quantify this influence of different sizes of the target region in the inpainting process. So we artificially make missing squares in the image shown in Figure 8. Assuming the original image is the perfect inpainting result we want to obtain, we calculate the PSNR between original image and inpainted one.

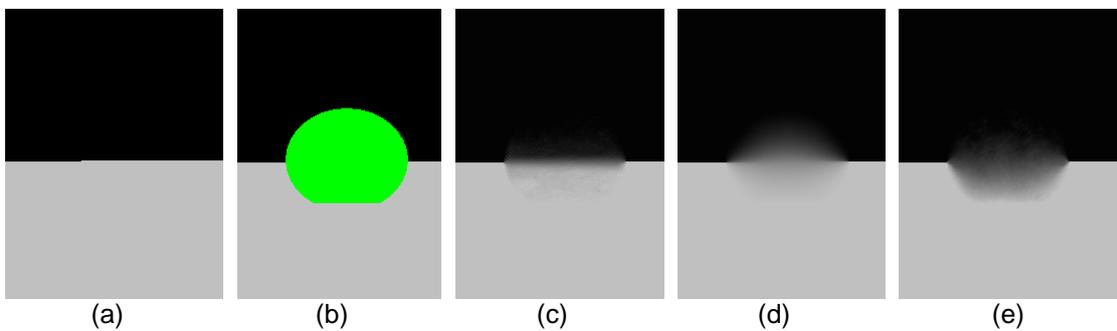


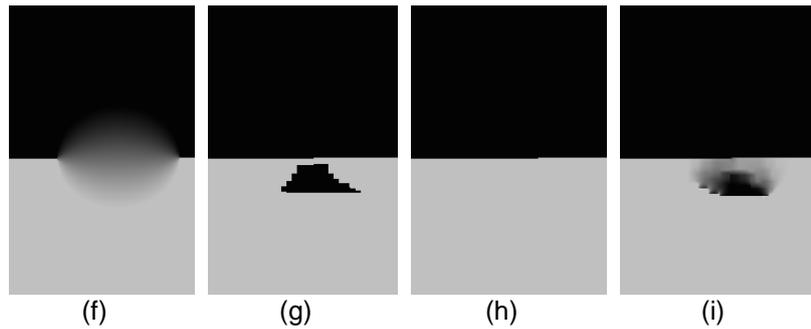
**Figure 8. Image Covered (Damaged) by 9 Blocks with Different Sizes**



**Figure 9. Profile of Different Algorithms Inpainting Blocks (Block Size is in 10, e.g., at 12, the Real Block Size is 120\*120). For the Variational Methods (BSCB, TV, CDD, Fast Inpainting), the Step Size and Iteration Time are Fixed as 1 and 2000 Separately. And for the Exemplar Based Methods, the Inpainting Patch Size is 11\*11**

For images with sharp edges and discontinuities, we select Figure 10 (a), and mask the inpainted area in green like Figure 10 (b). Different from Figure 8 (a), Figure 10 (a), is a synthetic image with two color blocks divided by one sharp edge in the middle. And Figure 10 (c)-(i), are the results of BSCB method [1], TV method [2], CDD method [4], fast inpainting [17], Criminisi method [6], nonlocal mean method [32] and Poisson blending [35] inpainting. Computing time of each algorithm is provided in Table 1.



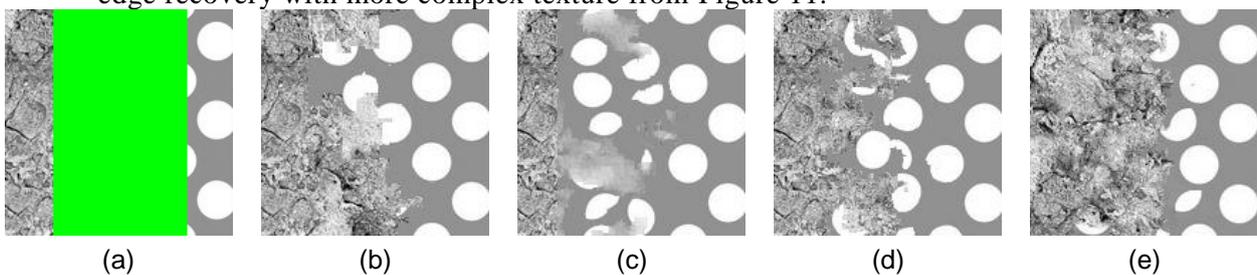


**Figure 10. Result of Different Variational Algorithms for an Image with Edge Structure. (a) Original Image; (b) Masked Image; (c) BSCB Method; (d) TV Method; (e) CDD Method; (f) Fast Inpainting; (g) Criminisi Method; (h) Nonlocal Mean Method; (i) Poisson Blending**

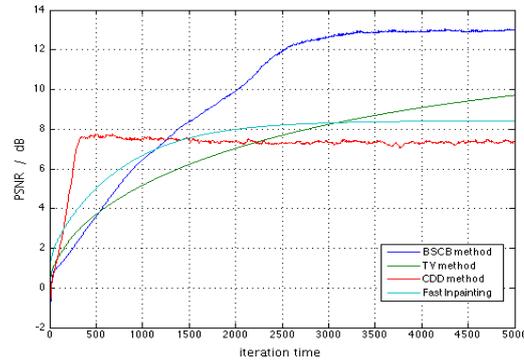
**Table 1. Overview of the Inpainting Results of Different Algorithms**

PDE methods	PSNR(dB)	TIME(s)	Step size	Exemplar-based methods	PSNR(dB)	TIME(s)	Patch size
BSCB method	12.8	27.8	1.25	Criminisi	4.8	8.8	11
TV method	9.8	31.7	1	Non-local mean	19.2	11.2	11
CDD method	7.4	361.8	1.25	Poisson	3.6	29.8	11
Fast Inpainting	8.4	18.5	1	Non-local median	34.722	58.9	11

Figure 10, shows that TV method has a great advantage to keep the edge line over other PDE methods. Also, due to the problem shown in Figure 4, results of Criminisi method and Poisson blending have an obvious artifact. In Figure 12 and Table 1. PSNR evaluation of different variational algorithms and computing time are provided. Additionally, we conduct further experiment on four exemplar-based methods about edge recovery with more complex texture from Figure 11.

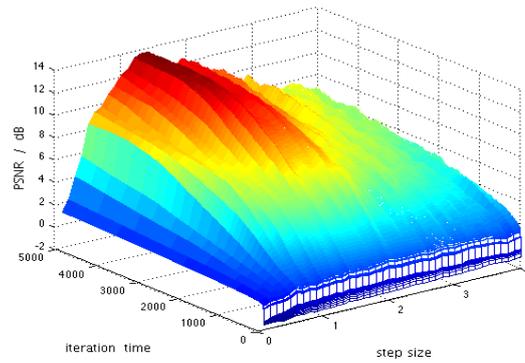


**Figure 11. Result of 4 Exemplar based Algorithms. The Mask to be Inpainted is the Green Bar. (a) Masked Image; (b) Result of Criminisi Algorithm; (c) Result of NL-MEAN Algorithm; (d) Result of NL-MEDIAN Algorithm and (e) Result of POISSON Algorithm. (The Patch Size is 11\*11)**



**Figure 12. Evaluation of Different Variational Algorithms**

The step size is crucial in practical applications. Using the method in [1], we change the step from 0.01 to 4, and show the PSNR results in Figure 13. When the step size is too small or too big, the result would converge in a very low PSNR.



**Figure 13. Inpainting Result from Different Step Size.**

A specific application of image inpainting is inpainting text. In our experiment, we only consider visual acceptability regardless of the semantic meaning of the results. Inpainted area cannot be blurred (Figure 14 (c)), disordered (Figure 14 (e)(f)), or even with too much blank (Figure 14 (d)). The best result is from Poisson method or Criminisi method, presented in Figure 14.

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

(a) Original image

(b) image with missing area

(c) PDE method

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

(d) Criminisi method

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

(e) Non-local mean

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

(f) Non-local median

Non-local methods in inpainting have gained popularity due to the work of [11], although the authors claim the inspiration from Efros and Leung [3]. Authors of [13] have used graph differential geometry to provide a discrete formulation of image processing methods such as bilateral filter, the TV digital inhomogeneous regional means filter. The advantage of choosing graphs is that they can model well the interactions between data. The local or the nonlocal regularization can be derived by the way the connections are made in the graph.

Let  $I$  be the Image whose domain is  $\Omega$  in  $\mathbb{R}^2$ . Let  $\Phi$  be an embedding that maps each pixel to a square patch. The area of each patch is considered to be  $d$ .  $\Phi(p)$ ,  $\Phi(q)$  represent square patches around the pixels  $p$ ,  $q$  respectively. They are  $d$ -tuples obtained by lexicographically arranging the pixels inside the corresponding patches.

Let  $G = (V, E, w)$  be a weighted graph.  $V$  and  $E$  are the set of vertices and edges respectively. We connect two vertices  $p, q$

(g) Poisson method

Figure 14. Example of Text Inpainting

From the results above, exemplar-based methods generally have better performance over PDE methods. However, there is one type of image impairment that exemplar-based methods cannot finish. Target region in applications such as image super-resolution demosaicing and zooming consists of sub-regions with trivial size but in enormous quantity. In such situation, exemplar-based methods cannot find a single block in which there is no unknown pixel. Results are shown below.



Figure 15. Images Covered by Gaussian Noise with Different Percentage

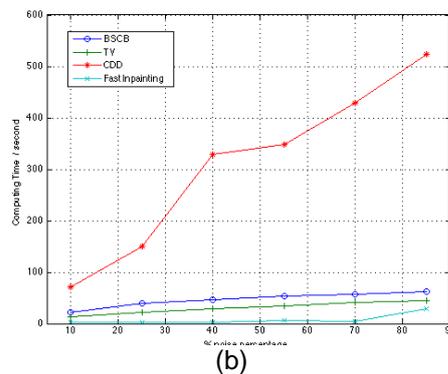
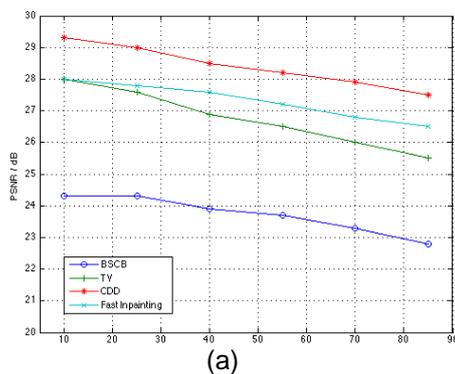


Figure 16. Profile of Different Algorithms Inpainting Global Gaussian Noise. (a) Relation between Noise Percentage and PSNR of Different Algorithms; (b) Relation between Noise Percentage and Computing Time

## 6. Conclusion

A comprehensive study about image inpainting techniques is proposed in this paper. Different algorithms are discussed and compared. Since both variational methods and exemplar-based method have their own limitation, in our opinion, a good algorithm should take both the structure and the texture into consideration, for example, successful avant-garde works in [52-55]. In the last decade, deep learning has been highly developed, and has shown a great power on dealing with complex classification and regression problems. One alternative could be to use deep neural network (such as GRNN) to make regression on the unknown pixels. In fact the fast inpainting in [17] can be viewed as a one-layer local-connected neural network. Out future work will focused on how to create a valid training dataset and how to choose a proper network structure for different inpainting problems.

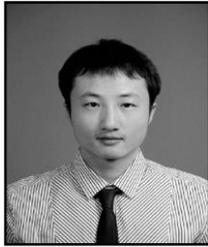
## References

- [1] M. Bertalmio, G. Sapiro, V. Caselles and C. Ballester, "Image inpainting", In Proceedings of ACM SIGGRAPH, vol. 4, (2000), pp. 417-424.
- [2] T. Chan and J. Shen, "Local inpainting models and tv inpainting", SIAM J. Appl. Math, vol. 62, no. 3, (2001), pp. 1019-1043.
- [3] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels", IEEE Trans. Image Processing, vol. 10, (2001), pp. 1200-1211.
- [4] T. F. Chan and J. Shen, "Nontexture inpainting by curvature-driven diffusions", Journal of Visual Communication & Image Representation, vol. 12, no. 4, (2001), pp. 436-449.
- [5] K. Toyama, P. Perez and A. Criminisi, "Object Removal by Exemplar-Based Inpainting", 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, vol. 2, pp. 721, (2003).
- [6] A. Criminisi, P. Perez and K. Toyama, "Region filling and object removal by exemplar-based image inpainting", Image Processing IEEE Transactions on, vol. 13, no. 9, (2004), pp. 1200-1212.
- [7] O. L. Meur and C. Guillemot, "Super-resolution-based inpainting", Lecture Notes in Computer Science, vol. 7577, no. 1, (2012), pp. 554-567.
- [8] J. Verdera, V. Caselles, M. Bertalmio and G. Sapiro, "Inpainting surface holes. Image Processing, 2003. ICIP 2003", Proceedings. 2003 International Conference on, IEEE, vol. 2, (2003), pp. 903-906.
- [9] J. Shen, "Inpainting and the fundamental problem of image processing", Siam Review, (2003).
- [10] S. Esedoglu and J. Shen, "Digital inpainting based on the mumford-shah-euler image model", European Journal of Applied Mathematics, vol. 13, no. 3, (2002), pp. 353-370.
- [11] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems 69", Communications on Pure & Applied Mathematics, vol. 42, no. 5, (1989), pp. 577-685.
- [12] M. Bertalmio, A. L. Bertozzi and G. Sapiro, "Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting", 2013 IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, vol. 1, (2001), pp. 355.
- [13] A. E. And, A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling", International Conference on Computer Vision, vol. 2, (1999), pp. 1033-1038.
- [14] T. F. Chan and J. Shen, "Mathematical Models for Local Nontexture Inpaintings", SIAM J. Appl. Math vol. 62, (2002), pp. 1019-1043.
- [15] G. Kanizsa, "Organization in vision praeger", n.y. [2] von der, America, (1979) May.
- [16] M. Nitzberg, D. Mumford and T. Shiota, "Filtering, segmentation, and depth", Springer Berlin, (1993).
- [17] M. M. Oliveira, B. Bowen, M. Richard and Y. Chang, "Fast digital image inpainting", Proc Viip, (2001), pp. 261-266.
- [18] J. H. Choi and C. H. Hahm, "An exemplar-based image inpainting method with search region prior", Consumer Electronics (GCCE), 2013 IEEE 2nd Global Conference on, (2013), pp. 68-71.
- [19] K. Sangeetha, P. Sengottuvelan and E. Balamurugan, "A novel exemplar based image inpainting algorithm for natural scene image completion with improved patch prioritizing", International Journal of Computer Applications, vol. 36, (2012).
- [20] H. Yamauchi, J. Hitoshi and H. P. Seidel, "Image restoration using multiresolution texture synthesis and image inpainting", Computer Graphics International, Proceedings, IEEE, (2003).
- [21] J. Sun, L. Yuan, J. Jia and H. Shum, "Image completion with structure propagation", ACM Transactions on Graphics (ToG) , ACM, vol. 24, no. 3, (2005).
- [22] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images", Journal of Applied Statistics, vol. 6, no. 6, (1987), pp. 564-584.
- [23] P. Brémaud, "Markov chains: gibbs fields, monte carlo simulation, and queues", Technometrics, vol. 42, no. 4, (2000), pp. 438-439.
- [24] A. E. And, A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling", International Conference on Computer Vision, vol. 2, , (1999), pp. 1033-1038.

- [25] M. Ashikhmin, "Synthesizing Natural Textures", In ACM Symposium on Interactive 3D Graphics, (2001), pp. 217-226.
- [26] S. Lefebvre and H. Hoppe, "Appearance-space texture synthesis", *Acm Transactions on Graphics*, vol. 25, no. 3, (2006), pp. 541-548.
- [27] L. Demanet, B. Song and T. Chan, "Image inpainting by correspondence maps: a deterministic approach", *Variational Level Set Methods, Prod. Of Workshop in Int'l Conf. Image Proc.*, (2004).
- [28] J. Wang, K. Lu, D. Pan, N. He and B. K. Bao, "Robust object removal with an exemplar-based image inpainting approach", *Neurocomputing*, vol. 123, (2014), pp. 150-155.
- [29] Z. D. Li, H. He, Z. Yin and F. Chen, "A robust and adaptive image inpainting algorithm based on a novel structure sparsity", *International conference on image processing, computer vision & pattern recognition (IPCV) Las Vegas, American*, vol. 1, (2013), pp. 328-333.
- [30] Anupam, P. Goyal and S. Diwakar, "Fast and Enhanced Algorithm for Exemplar Based Image Inpainting", *Image and Video Technology (PSIVT), 2010 Fourth Pacific-Rim Symposium on, IEEE*, (2010), pp. 325-330.
- [31] Y. F. Liu, F. L. Wang and X. Y. Xi, "Enhanced Algorithm for Exemplar-Based Image Inpainting", 2013, Ninth International Conference on Computational Intelligence and Security, IEEE, (2013), pp. 209-213.
- [32] A. Wong and J. Orchard, "A nonlocal-means approach to exemplar-based inpainting", *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on, IEEE* (2008), pp. 2600-2603.
- [33] M. Daisy, D. Tschumperlé and O. Lézoray, "Spatial patch blending for artefact reduction in pattern-based inpainting techniques", *Lecture Notes in Computer Science*, (2013), pp. 523-530.
- [34] M. Daisy, D. Tschumperlé and O. Lézoray, "A fast spatial patch blending algorithm for artefact reduction in pattern-based image inpainting", *Siggraph Asia Technical Briefs*, (2013).
- [35] H. M. Nguyen, B. C. Wünsche, P. Delmas and C. Lutteroth, "Poisson blended exemplar-based texture completion", *Proceedings of the Thirty-Seventh Australasian Computer Science Conference - Volume 147 (Vol.147, pp.99-104). Australian Computer Society, Inc.*, (2014).
- [36] P. Pérez, M. Gangnet and A. Blake, "Poisson image editing", *Acm Siggraph*, vol. 22, no. 22, (2003), pp. 313-318.
- [37] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang and S. Yan, "Sparse representation for computer vision and pattern recognition", *Proceedings of the IEEE*, vol. 98, no. 6, (2010), pp. 1031-1044.
- [38] M. Elad, J. Starck, L. P. Querre and D. L. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis", *MCA, Journal on Applied and Computational Harmonic Analysis*, (2005).
- [39] S. Chen and D. Donoho, "Atomic decomposition by basis pursuit", *Siam Review*, vol. 43, no. 1, (1998), pp. 33-61.
- [40] G. M. Davis, S. G. Mallat and Z. Zhang, "Adaptive time-frequency decompositions", *Optical Engineering*, vol. 33, no. 7, (1994), pp. 2183-2191.
- [41] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries", *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, (1993), pp. 3397-3415.
- [42] Y. C. Pati, R. Rezaeiifar and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition", In *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, (1993), pp. 1-3.
- [43] D. L. Donoho, Y. Tsaig, I. Drori and J. L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit", *IEEE Transactions on Information Theory*, vol. 58, no. 2, (2012), pp. 1094-1121.
- [44] D. Needell and J. A. Tropp, "Cosamp: iterative signal recovery from incomplete and inaccurate samples", *Applied and Computational Harmonic Analysis*, vol. 26, no. 12, (2009), pp. 301-321.
- [45] M. Aharon, "K-svd: an algorithm for designing overcomplete dictionaries for sparse representation.", *IEEE Trans on Signal Processing*, vol. 54, (2006).
- [46] M. J. Fadili, J. L. Starck and F. Murtagh, "Inpainting and zooming using sparse representations", *The Computer Journal*, vol. 52, no. 1, (2009), pp. 64-79.
- [47] B. Shen, W. Hu, Y. Zhang and Y. J. Zhang, "Image inpainting via sparse representation. Acoustics, Speech, and Signal Processing", *IEEE International Conference on* (2009), pp. 697-700.
- [48] Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity", *Image Processing IEEE Transactions on*, vol. 19, no. 5, (2010), pp. 1153-1165.
- [49] M. J. Fadili and J. L. Starck, "Em Algorithm For Sparse Representation-Based Image Inpainting", *Image Processing, IEEE International Conference on*, II-61-4, (2005).
- [50] P. P. Shi, Q. S. Lian and Q. Shang, "Image inpainting algorithm based on three-layer sparse representation", *Computer Engineering*, (2010).
- [51] Y. J. Liu and C. Shu, "A comparison of image inpainting techniques", *Sixth International Conference on Graphic and Image Processing, ICGIP*, (2014).
- [52] P. Arias, G. Facciolo, V. Caselles and G. Sapiro, "A variational framework for exemplar-based image inpainting", *International Journal of Computer Vision*, vol. 93, no. 3, (2011), pp. 319-347.
- [53] J. Wu and Q. Ruan, "A novel exemplar-based image completion model", *Journal of Information & Engineering*, vol. 25, no. 2, (2009), pp. 481-497.

- [54] F. Cao, Y. Gousseau, S. Masnou and P. Pérez, “Geometrically guided exemplar-based inpainting”, *SIAM Journal on Imaging Sciences*, (2011).
- [55] O. L. Meur, J. Gautier and C. Guillemot, “Exemplar-based inpainting based on local geometry”, *Image Processing (ICIP 2011)*, 18th IEEE International Conference on, (2011), pp. 3401-3404.

## Authors



**Chang Shu**, he received his PhD degree in Information and Communication Engineering from Tsinghua University in 2011. He is currently an assistant professor in the School of Communication and Information Engineering, the University of Electronic Science and Technology of China. His research interests include image processing, pattern recognition and computer vision.



**Yaojie Liu**, he received his Bachelor's degree from the University of Electronic Science and Technology of China in 2014. He is currently studying for a master's degree at the Ohio State University in the United States. His research interests include image processing and computer vision.

