



# Image Inpainting: A Review

Omar Elharrouss<sup>1</sup> · Noor Almaadeed<sup>1</sup> · Somaya Al-Maadeed<sup>1</sup> · Younes Akbari<sup>1</sup>

Published online: 6 December 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

Although image inpainting, or the art of repairing the old and deteriorated images, has been around for many years, it has recently gained even more popularity, because of the recent development in image processing techniques. With the improvement of image processing tools and the flexibility of digital image editing, automatic image inpainting has found important applications in computer vision and has also become an important and challenging topic of research in image processing. This paper reviews the existing image inpainting approaches, that were classified into three subcategories, sequential-based, CNN-based, and GAN-based methods. In addition, for each category, a list of methods for different types of distortion on images are presented. Furthermore, the paper also presents available datasets. Last but not least, we present the results of real evaluations of the three categories of image inpainting methods performed on the used datasets, for different types of image distortion. We also present the evaluations metrics and discuss the performance of these methods in terms of these metrics. This overview can be used as a reference for image inpainting researchers, and it can also facilitate the comparison of the methods as well as the datasets used. The main contribution of this paper is the presentation of the three categories of image inpainting methods along with a list of available datasets that the researchers can use to evaluate their proposed methodology against.

**Keyword** Image inpainting · Objects removal · Image repairing · CNN · GAN

---

✉ Omar Elharrouss  
elharrouss.omar@gmail.com

Noor Almaadeed  
n.alali@qu.edu.qa

Somaya Al-Maadeed  
s\_alali@qu.edu.qa

Younes Akbari  
akbari\_younes@semnan.ac.ir

<sup>1</sup> Department of Computer Science and Engineering, Qatar University, Doha, Qatar

## 1 Introduction

Nowadays, image and video are the most common forms of information needed in different domains of life [1,2], and the processing of the content of the images/videos become a challenging task because of the informations that can be in it [3,4]. The extraction of such information is related to the purpose of the analysis [5–7]. In addition, it is a crucial tool for monitoring the security of people and objects [8–12]. But editing applications that can edit an image without leaving any traces, pose a problem to the public trust and confidence. Thus, the need for an automatic system to detect and extract the real image present in the available is an urgent demand. Meanwhile, the availability of original image from the given image is heavily dependent on the extraction mechanism of the original image; hence, object removal from images is one of the big concerns of the research and a hot topic for information security [13,14].

Shared images in the social networks can contain many objects added to these images including signature, rectangles or emoticons. The addition of these objects can change the semantic of images. Removing these objects from the images is a widely recognized problem and a current track in computer vision research. Also, object removal is considered a solution for forgery of images. Object removal techniques that exist in literature can be divided into two categories: image inpainting and copy-move methods. The copy move based methods perform the undesired object removal by extracting a part from another image or another region from the same image, then pasting it to the object region that we are trying to remove. This technique is widely used for object removal due its simplicity, however; it is not suitable for some cases like face images or complicated scenes. Image inpainting, on the other hand, was applied on old images in order to remove scratches and enhance damaged images. Now, it is used for removing artifact objects that can be added to images by filling the target region with estimated values. Image inpainting is also used to remove any type of distortion including text, blocks, noise, scratch, lines or many types of masks [15–17]. Figure 1 represents the different existing types of distortion. By using recently developed algorithms, image inpainting can restore coherently both texture and structure components of the image. The obtained results demonstrate that these methods can remove undesirable objects from the images without leaving traces like artifacts ghosts. Until now, few methods are proposed for blind image inpainting regarding the massive number of published works with different techniques like sequential-based, CNN-based or GAN-based.

Removing objects from images using image inpainting can reach improved performance in the future, but when the image editors hide traces using sophisticated techniques, the detection of forgery and the inpainting of image become difficult. For that reason, almost all detection approaches attempt to handle this by detecting the abnormalities of similarity between blocks of the image that can be affected during the postprocessing operation. This work summarizes different methods for image inpainting using different techniques including sequential-based, CNN-based or GAN-based methods.

The remainder of the paper is organized as follows: the literature overview including sequential-based, CNN-based and GAN-based methods is presented in Sect. 2. Employed datasets are presented in Sect. 3. Evaluations and metrics used are discussed in Sect. 4. The conclusion is provided in Sect. 5.

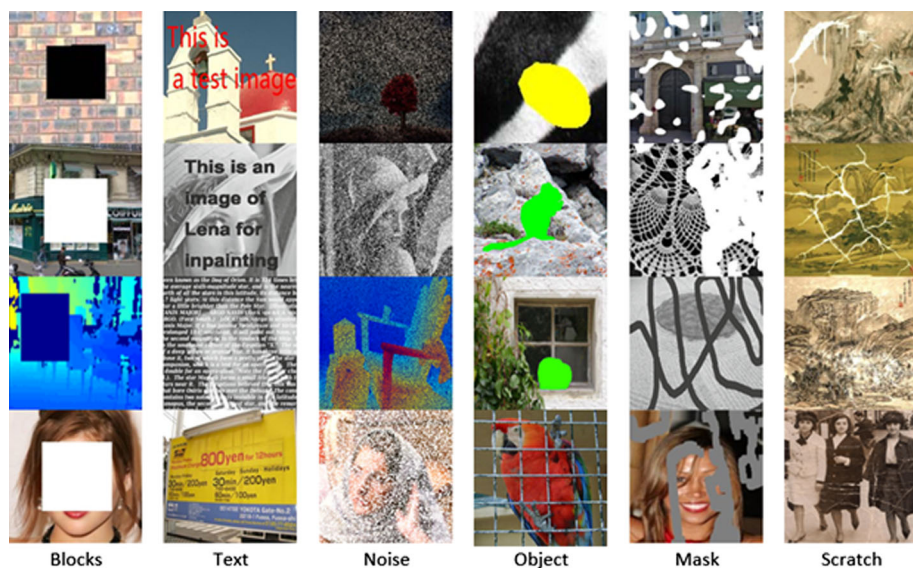


Fig. 1 Types of distortion

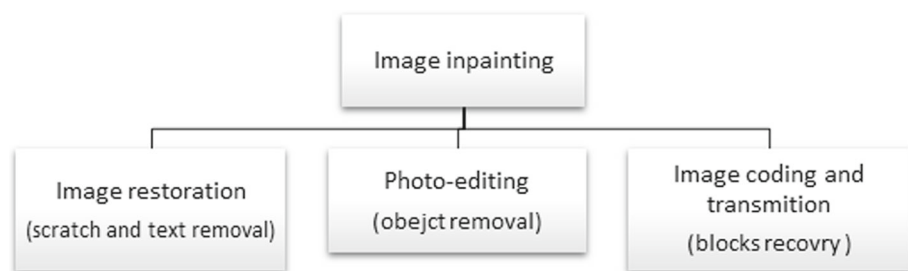


Fig. 2 Image inpainting applications and the purposes of each category

## 2 Literature Review

Image inpainting is the process of completing or recovering the missing region in the image or removing some objects added to it. The operation of inpainting depends on the type damaging in the image, and the application that caused this distortion. For example, in image restoration, we talk about removing the scratch or text that can be found in the images, whereas, in a photo-editing application, we are interested in object removal in image coding and transmission applications, the operation related to images inpainting is recovering the missing blocks. Finally, for virtual paintings restoration, the related operation is scratch removal. Figure 2 is a representation of the kind of application and the corresponding image inpainting operation.

To handle this, many methods have been proposed including sequential algorithms or deep learning techniques. For that, we categorize the existing methods for images inpainting into three categories: sequential-based approaches, CNN-based approaches, and GAN-based approaches. The sequential-based approaches are the proposed method without deep learning using neural networks. Where the CNN-based approaches are the algorithms that use neural networks with automatically deep learning. GAN-based approaches represent the

methods that use Generative adversarial networks (GANs) for training the images of inpainting models.

By the following, the image inpainting works related to each category of methods is presented.

## 2.1 Sequential-Based Methods

Approaches related to images inpainting can be classified into two categories: patch-based and diffusion-based methods.

Patch-based methods are based on techniques that fill in the missing region patch-by-patch by searching for well-matching replacement patches (i.e., candidate patches) in the undamaged part of the image and copying them to corresponding locations. Many methods have been proposed for image inpainting using patch-based method. Ružić and Pižurica [15] proposed a patch-based method consisting of searching the well-matched patch in the texture component using Markov random field (MRF). Jin and Ye [16] proposed a patch-based approach based on annihilation property filter and low rank structured matrix. In order to remove an object from an image, Kawai et al. [17] proposed an approach based on selecting the target object and limiting the search around the target by the background. Using two-stage low rank approximation (TSLRA) [18] and gradient-based low rank approximation [19], authors proposed patch-based methods for recovering the corrupted block in the image. On RGB-D images full of noise and text, Xue et al. [20] proposed a depth image inpainting method based on Low Gradient Regularization. Liu et al. [21] used the statistical regularization and similarity between regions to extract dominant linear structures of target regions followed by repairing the missing regions using Markov random field model (MRF). Ding et al. [22] proposed a patch-based method for image inpainting using Nonlocal Texture Matching and Nonlinear Filtering (Alpha-trimmed mean filter). Duan et al. [23] proposed an image inpainting approach based on the Non-Local Mumford–Shah model (NL–MS). Fan and Zhang [24] proposed another image inpainting method based on measuring the similarity between patches using the Sum of Squared Differences (SSD). In order to remove blocks from an image, Jiang [25] proposed a method for image compression. Using Singular value decomposition and an approximation matrix, Alilou and Yaghmaee [26] proposed an approach to reconstruct the missing regions. Other notable research includes using texture analysis on Thangka images to recover missing block in an image [27] and using the structure information of images [28,29]. In the same context, Zeng et al. [30] proposed the use of Saliency Map and Gray entropy. Zhang et al. [31] proposed an image inpainting method using a joint probability density matrix (JPDM) for object removal from images.

Wali et al. [32] proposed a denoising and inpainting method using total generalized variation (TGV). The authors analyze three types of distortion including text, noise, masks. In the same context, Zhang et al. [33] proposed an example-based image inpainting approach based on color distribution by restoring the missed regions using the neighboring regions. This work analyses the many types of distortions including objects, text, and scratch. The multiscale graph cuts technique is used for inpainting images in [34] by analyzing different types of distortion. In [35] the authors proposed a novel joint data-hiding and compression scheme for digital images using side match vector quantization (SMVQ) and image inpainting. The proposed approach is tested on six grayscale recognized images including Lena, airplane, peppers, sailboat, lake, and Tiffany. In order to preserving the texture consistency and structure coherence, the authors in [36] remove the added objects in the images using multiple pyramids method, local patch statistics and geometric feature-based sparse representation.

For 3D stacked image sensor the authors in [37] proposed an image inpainting method using discrete wavelet transform (DWT). In order to fill the missed region, the authors in [38] proposed a patch-based method by search and fills-in these regions with the best matching information surrounding it. In the goal to reconstruct the Borehole images the authors in [39] proposed a method by analyzing the texture and structure component of the images. Helmholtz equation is used for inpainting images in [40], after the inpainting of the missed region the authors proposed a method for enhancing the quality of the images.

Diffusion-based methods fill in the missing region (i.e. hole), by smoothly propagating image content from the boundary to the interior of the missing region. For that, Li et al. [41] proposed a diffusion-based method for image inpainting by localizing the diffusion of inpainted regions followed by the construction of a feature set based on the intra-channel and inter-channel local variances of the changes to identify the inpainted regions. Another diffusion-based method of image inpainting proposed by the same authors in a later research [42] involves exploiting diffusion coefficients which were computed using the distance and direction between the damaged pixel and its neighborhood pixel. Sridevi et al. [43] proposed another diffusion-based image inpainting method based on Fractional-order derivative and Fourier transform. Table 1 depicts a summary of patch-based and diffusion-based sequential methods for image inpainting.

Jin et al. [44] proposed an approach called sparsity-based image inpainting detection based on canonical correlation analysis (CCA). Mo and Zhou [45] present a research-based on dictionary learning using sparse representation. These methods are robust for simple images, but when the image is complex like contains a lot of texture and object or the object cover a large region in the images, searching for similar patch can be difficult.

## 2.2 Convolutional-Neural-Network-Based Methods

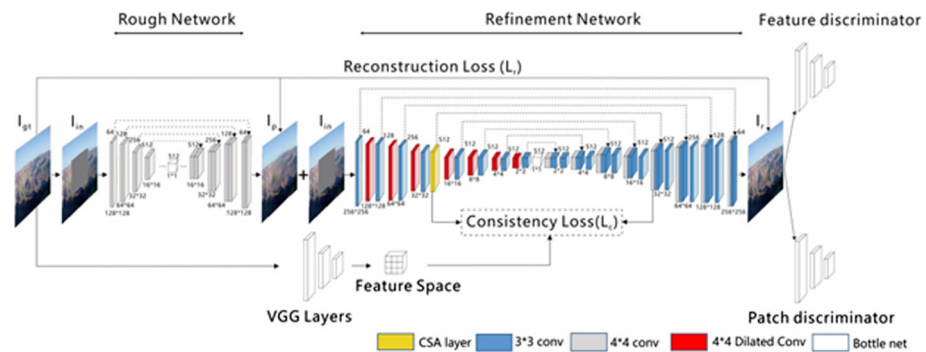
Recently, the strong potential of deep convolutional networks (CNNs) is being exhibited in all computer vision tasks, especially in image inpainting. CNNs are used specifically in order to improve the expected results in this field using large-scale training data. The sequential-based methods succeed in some parts of image inpainting like filling texture details with promising results, yet the problem of capturing the global structure is still a challenging task [46]. Several methods have been proposed for image inpainting using convolutional neural networks (CNNs) or encoder-decoder network based on CNN. Shift-Net based on U-Net architecture is one of these methods that recover the missing block with good accuracy in terms of structure and fine-detailed texture [46]. In the same context, Weerasekera et al. [47] use depth map of the image as input of the CNN architecture, whereas Zhao et al. [48] use the proposed architecture for inpainting X-ray medical images. VORNet [49] is another CNN-based approach for video inpainting for object removal. Most image inpainting methods know the reference of damaged pixels of blocks. Cai et al. [50] proposed a blind image inpainting method named (BICNN). Based on convolutional neural networks (CNNs) using encoder-decoder network structure many works have been proposed for image inpainting. Zhu et al. [51] proposed a patch-based inpainting method for forensics images. Using the same technique of encoder-decoder network, Sidorov and Hardeberg [52] proposed an architecture for denoising, inpainting, and super-resolution for noised, inpainted and low-resolution images, respectively. Zeng et al. [53] built a pyramidal-context architecture called PEN-NET for high-quality image inpainting. Liu et al. [54] proposed a layer to the encoder-decoder network called coherent semantic attention (SCA) layer for image inpainting method. This proposed architecture is presented in Fig. 3. Further, Pathak et al.

**Table 1** Sequential-based method for image inpainting

Category	Method	Feature	Image
Patch-based	Muddala et al. [13]	Layered depth image (LDI)	RGB
	ISOAWA et. [14]	Super-pixel	RGB
	Ružić et al. [15]	Markov random field (MRF)	RGB
	Liu et al. [21]		
	Jin et al. [16]	Annihilation property filter, low rank structured matrix	RGB
	Kawai et al. [17]	Background geometry estimation	Artificial
	Guo et al. [18]	Two-stage low rank approximation (TSLRA)	RGB
	Xue et al. [20]	Low Gradient Regularization	Depth
	Ding et al. [22]	Nonlocal Texture Matching, Nonlinear Filterin ( $\alpha$ -trimmed mean filter)	RGB
	Duan et al. [23]	Non-local Mumford–Shah model (NL–MS)	RGB
	Fan et al. [24]	Sum of Squared Differences (SSD)	Gray scale
	Jiang et al. [25]	Canny filter, Segmentation	RGB
	Alilou et al. [26]	Singular value decomposition and an approximation matrix	RGB
	Lu et al. [19]	Gradient-based low rank approximation	Gray scale
	Wang et al. [27]	Structure and texture analysis	Thangka images
	Yao et al. [29]		
	Wei et al. [28]	Structure-aware	RGB
	Zeng et al. [30]	Saliency Map and Gray entropy	RGB
	Zhang et al. [31]	Joint probability density matrix (JPDM).	RGB
	Wali et al. [32]	Total generalized variation (TGV)	RGB, Gray scale
	Liu et al. [34]	Multiscale graph cuts	RGB
	Ghorai et al. [36]	Local Patch Statistics and Steering Kernel Feature	RGB
Diffusion-based	Gao et al. [37]	Discrete wavelet transform (DWT)	Gray scale
	Li et al. [41]	Intra-channel and Inter-channel local variances	RGB
	Li et al. [42]	Distance and direction between the damaged pixel and its neighborhood pixels.	RGB
	Sridevi et al. [43]	Fractional-order derivative and Fourier transform	Gray scale

[55] proposed encoder-decoder model for image inpainting. In order to fill the gap between lines drawing in an image, Sasaki et al. [56] used an encoder-decoder-based model. This work can be helpful for scanned data that can miss some parts. For the UAV data that can be affected in terms of resolution or containing some blindspots, Hsu et al. [57] proposed a solution using VGG architecture. Also, for removing some text from the images Nakamura et al. [58] proposed a text erasing method using CNN. In order to enhance the images of the damaged artwork, Xiang et al. [59] also proposed a CNN-based method. In the same context as [59] and using GRNN neural network, Alilou and Yaghmaee [60] proposed a non-texture





**Fig. 3** Encoder-decoder networks model in [54]

**Table 2** CNN-based method for image inpainting

Method	Architecture	Data type
Yan et al. [46], Chang et al. [49], Zhu et al. [51], Liu et al. [54], Hsu et al. [57], Liao et al. [61], Hertz et al. [63], Zhao et al. [64], Su et al. [65], Ma et al. [66]	Encoder-decoder network	RGB
Weerasekera et al. [47]	CNN	TGB-D
Zhao et al. [48]	CNN	Y-ray
Cai et al. [50]	Blind CNN	Grayscale
Sidorov et al. [52]	3D CNN	RGB
Zeng et al. [53]	Pyramid-context encoder	RGB
Pathak et al. [55]	Context-encoder, CNN	RGB
Sasaki et al. [56]	CNN	Line drawing images
Nakamura et al. [58]	CNN	Image with text
Xiang et al. [60]	CNN, GAN	Damaged old pictures
Cai et al. [62]	CNN	RGB
Ke et al. [67]	U-Net and ResNet	RGB
Liu et al. [68]	Encoder-Interleaved Zooming Network-decoder	RGB
Guo et al. [69]	Full-resolution residual block (FRRB)	RGB

image inpainting method. Unlike the previous methods, Liao et al. [61] proposed a method called Artist-Net for image inpainting. The same goal is reached by Cai et al. [62] who proposed a semantic object removal approach using CNN architecture. In order to remove motifs from single images, Hertz et al. [63] proposed a CNN-based approach. Table 2 summarizes the CNN-based methods with a description of the type of data used for image inpainting.

For the same purpose of image inpainting, but for replacing a region of an image by another region from another image, the authors in [64] based on VGG model trained their own model. In order to mitigate the effect of the gradient disappearance, the authors in [65]

introduce a dense block for U-Net architecture that is used for inpainting the images. For medical purposes, the authors in [67] attempted to denoising the medical images using the principle of image inpainting using Residual U-Net architecture. To address the blurring and color discrepancy problems for image inpainting the authors in [66] proposed a method for missed region reconstruction using region-wise convolutions. As the authors in [68] add some layers named Interleaved Zooming Block in the encoder-decoder architecture for inpainting the images. The authors in [69] Proposed a full-resolution residual block (FRRB) with an encoder-decoder model for the same purpose.

## 2.3 GAN-Based Methods

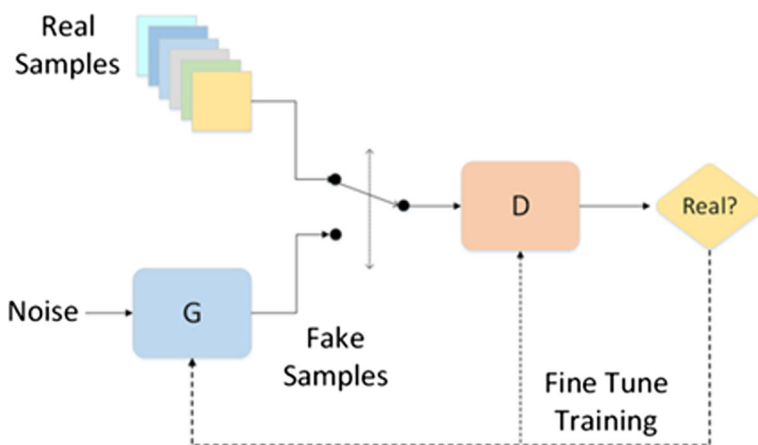
The much-used technique nowadays, was introduced for image generation in 2014 in [70]. Generative adversarial networks (GANs) are a framework which contains two feed-forward networks, a generator G and a discriminator D. The generative network, G, is trained to create a new image which is indistinguishable from real images, whereas a discriminative network, D is trained to differentiate between real and generated images. This relation can be considered as a two-player min-max game in which G and D compete. To this end, the G (D) tries to minimize (maximize) the loss function, i.e. adversarial loss, as follows:

$$\min_G \max_D E_{x \sim P_{data}(x)} [\log D(x)] + E_{z \sim P_z(z)} [\log(1 - D(G(z)))] \quad (1)$$

where  $z$  and  $x$  denote a random noise vector and a real image sampled from the noise  $P_z(z)$  and real data distribution  $P_{data}(x)$ , respectively. Recently, the GAN has been applied to several semantic inpainting techniques in order to complete the hole region naturally.

GANs are a framework that contains two feed-forward networks, a generator G and a discriminator D, as shown in Fig. 4. The generator takes random noise  $z$  as input and generates some fake samples similar to real ones; while the discriminator has to learn to determine whether samples are real or fake. At present, Generative Adversarial Network (GAN) becomes the most used technique in all computer vision applications. GAN-based approaches use a coarse-to-fine network and contextual attention module gives good performance and is proven to be helpful for inpainting [71–75]. Existing image inpainting methods based on GAN are generally a few. Out of these, we find that in [71], Chen and Hu proposed a GAN-based semantic image inpainting method, named progressive inpainting, where a pyramid strategy from a low-resolution image to a higher one is performed for repairing the image. For handwritten images, Li et al. [72] proposed a method for inpainting and recognition of occluded characters. The methods use improved GoogLeNet and deep convolutional generative adversarial network (DCGAN). In an image inpainting method named PEPSI [76] the authors unify the two-stage cascade network of the coarse-to-fine network into a single-stage encoder-decoder network. Where PEPSI++ is the extended version of PEPSI [73]. In [74] the authors used Encoder-decoder network and multi-scale GAN for image inpainting. The same combination is used in [75] for image inpainting and image-to-image transformation purposes. On the RGB-D images, Dhama et al. [77] used CNN and GAN model to generate the background of a scene by removing the object in the foreground image as performed by many methods of motion detection using background subtraction [78,79]. In order to complete the missing regions in the image, Vitoria et al. [80] proposed an improved version of the Wasserstein GAN with the incorporation of Discriminator and Generator architecture. In the same context, but on sea surface temperature (SST) images, the Dong et al. [81] proposed a deep convolutional generative adversarial network (DCGAN) for filling the missing parts of the images. Also, Lou et al. [82] exploit a modifier GAN architecture for image inpainting



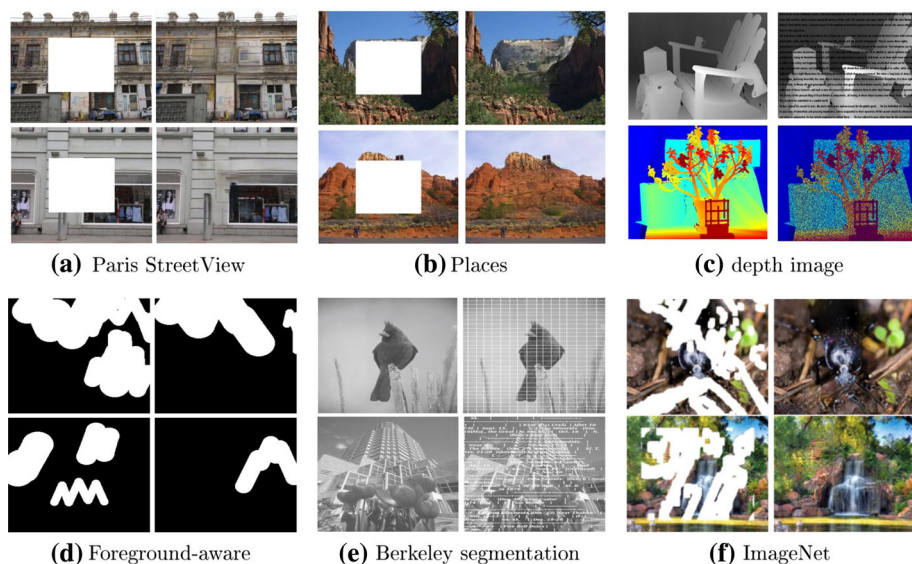


**Fig. 4** Framework of GANs

whereas, Salem et al. [83] proposed a semantic image inpainting method using adversarial loss and self-learning encoder-decoder model. A good image restoration method requires preserving structural consistency and texture clarity. For this reason, Liu et al. [84] proposed a GAN-based method for image inpainting on face images. FiNet [85] is another approach found in the literature for fashion image inpainting that consists of completing the missing parts in fashion images.

Recently, several approaches are proposed by combining some additional techniques (GAN, CNN,...) for inpainting the images. Jiao et al. [86] combined an encoder-decoder, multi-layer convolutions layers and GAN for restoring the images. The authors in [87] proposed a two-stage adversarial model named EdgeConnect by providing a generator for edge followed by an image inpainting model. The first model attempt to provide an edge completion component and the second one, inpaint the RGB image. According to the fact that GAN-based image inpainting models do not care out to the consistency of the structural and textural values between the inpainted region and their neighboring, the authors in [88] attempts to handle this limitation by providing a GAN model for learning the alignment between the block around the restored region and the original region. For the same reason as [88], taking into consideration the semantic consistency between restored images and original images, Li et al. [89] provided a boosted GAN model comprising an inpainting network and a discriminative network. When the inpainting network discovers the segmentation information of the input images, the discriminative network discovers the regularizations of the overall realness and segmentation consistency with the originals images. In the same context and using GAN-based models for images inpainting, each work provides some prior processing on GAN networks to get the best inpainting results for different types of images including medical images [90], face images [91] or scenes images [92].

The GAN-based methods give a good addition to the performance of image inpainting algorithms, but the speed of training is lower and needs very good performance machines, and this is due to computational resources requirements including network parameters and convolution operations.



**Fig. 5** Examples from Image inpainting datasets

### 3 Image Inpainting Datasets

Image inpainting methods use well known and large datasets for evaluating their algorithms and comparing the performance. The categories of images determine the effectiveness of each proposed method. These categories include natural images, artificial images, face images, and many other categories. In this work, we attempt to collect the most used datasets for image inpainting including Paris StreetView [93], Places [94], depth image dataset [20], Foreground-aware [95], Berkeley segmentation [96], ImageNet [97] and others. We also try to cite the types of used data such as RGB images, RGB-D images, and SST images. Figure 5 represents some frame examples from the cited datasets. Where Table 3 describes various datasets used for image inpainting approaches.

**Paris StreetView** [93] is collected from Google StreetView and represents a large-scale dataset containing street images for several cities around the world. The Paris StreetView comprises 15,000 images. The resolution of images is  $936 \times 537$  pixels.

**Places**<sup>1</sup> [94] dataset is built for human visual cognition and visual understanding purposes. The dataset contains many scene categories such as bedrooms, streets, synagogue, canyon, and others. The dataset is composed of 10 million images including 400+ images for each scene category. It allows the deep learning methods to train their architecture with large-scale data.

A **depth image** dataset<sup>2</sup> is introduced by Xue et al. [20] for evaluating depth image inpainting methods. The dataset is composed of two types of images: RGB-D images and grayscale depth images. Also, 14 scene categories are included such as Adirondack, Jade plant, Motorcycle, Piano, Playable and others. The masks for damaged images are created including textual makes (text in the images) and random missing masks.

<sup>1</sup> <http://places2.csail.mit.edu/download.htm>.

<sup>2</sup> <http://www.cad.zju.edu.cn/home/dengcai/Data/depthinpaint/DepthInpaintData.html>.

**Table 3** Datasets description

Dataset	Year	Content category	Image type	Size and resolution	Papers
Paris streetView	2015	Street	RGB	936 × 537	[25,33,34,40,41,45,46,56]
Places	2017	Miscellaneous	RGB	10Mil, 256 × 256	[25,30,32,45]
Foreground-aware	2019	Masks	Binary, grayscale	100K, 256 × 256	[7]
Berkeley segmentation	2001	Miscellaneous	RGB	12K	[4,10,29]
ImageNet	2015	Miscellaneous	RGB	100K, , various	[29,40,41,45]
CelebA	2018	Face images	RGB	200K, various	[43,45,52,56]
Indian Pines	2015	Scenes (agriculture, forest,...)	RGB, gray scale	145 × 145	[31]
Microsoft COCO val	2014	Miscellaneous	RGB	2.5 Mil, 256 × 256	[42]
ICDAR 2013	2013	RGB images with text	RGB	27 GB, various	[64]
SceneNet	2016	Miscellaneous	RGB, RGB-D	5Mil, various	[48]
Stanford Cars	2013	Cars	RGB	16K	[43]
Cityscapes	2016	Miscellaneous	RGB	48GB, various	[47]
Middlebury Stereo	2006, 2014	Miscellaneous	RGB, RGB-D, Grayscale depth	1396 × 1110 (2006), 2964 × 1988 (2014)	[7,8]

**Foreground-aware** dataset [95] is different from the other datasets. It contains the masks that can be added to any image for damaging it. It is named an irregular hole mask dataset for image inpainting. Foreground-aware datasets contain 100,000 masks with irregular holes for training, and 10,000 masks for testing. Each mask is a  $256 \times 256$  gray image with 255 indicating the hole pixels and 0 indicating the valid pixels. The masks can be added to any image for damaging it, which can be used for creating a large dataset of damaged images.

**Berkeley segmentation** database<sup>3</sup> [96] is composed of 12,000 images segmented manually. The images collected from other datasets consist of 30 human subjects. The dataset is a combination of RGB and Grayscale images.

**ImageNet**<sup>4</sup> [97] is a large-scale dataset with thousands of images of each subnets. Each subnet is represented by 1000 images. The current version of the dataset contains more than 14,197,122 images where the 1,034,908 images of the human body are annotated with a bounding box.

**USC-SIPI image** database<sup>5</sup> contains several volumes representing many types of images. The resolution in each volume can vary between  $256 \times 256$ ,  $512 \times 512$  and  $1024 \times 1024$  pixels. Generally, the datasets contain 300 images representing four volumes including texture, aerials, miscellaneous and sequences.

**CelebFaces Attributes** Dataset<sup>6</sup> (CelebA) [98] is a recognized and public datasets for face recognition. It contains more than 200,000 celebrity images representing 10,000 identities with a large pose variations.

**Indian Pines**<sup>7</sup> [99] consist of images representing images of three scenes including agriculture, forest and natural perennial vegetation with resolution of  $145 \times 145$  pixels.

**Microsoft COCO val2014** dataset<sup>8</sup> [100] is a new image recognition, segmentation, and captioning dataset. Microsoft COCO has several features with a total of 2.5 million labeled instances in 328,000 images.

**ICDAR 2013** dataset<sup>9</sup> [101] is a handwritten dataset with two languages including Arabic and English. The total number of writers is 475 whose handwritten page images have been scanned. The datasets contain 27 GB of data.

**SceneNet** dataset<sup>10</sup> [102] is a dataset for scene understanding tasks including semantic segmentation, object detection and 3D reconstruction. It contains RGB image and the corresponding RGB-D images, which form, in total, 5 million images.

**Stanford Cars** dataset<sup>11</sup> [103] is a set of car images representing 196 categories of cars of different sizes. The datasets contain 16,200 images in total.

**Cityscapes** dataset<sup>12</sup> [104] is a large-scale dataset of stereo videos of street scenes of 50 cities. The images contain about 30 classes of objects. Also, it contains about 20,000 annotated frames with coarse annotations.

<sup>3</sup> <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>.

<sup>4</sup> <http://image-net.org/>.

<sup>5</sup> <http://sipi.usc.edu/database/>.

<sup>6</sup> <http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html>.

<sup>7</sup> [http://www.ehu.eus/ccwintco/index.php/Hyperspectral\\_Remote\\_Sensing\\_Scenes](http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes).

<sup>8</sup> <http://cocodataset.org/#home>.

<sup>9</sup> <http://www.tamirhassan.com/html/competition.html>.

<sup>10</sup> <https://robotvault.bitbucket.io/scenenet-rgb-d.html>.

<sup>11</sup> <https://ai.stanford.edu/jkrause/cars/car/dataset.html>.

<sup>12</sup> <https://www.cityscapes-dataset.com/>.

**Middlebury Stereo**<sup>13</sup> datasets contain many versions we present the two new ones [105] and [106]. Middlebury 2006 [105] is a depth grayscale dataset that contains images captured from 7 view with different illuminations and exposures. The images resolution is defined by three categories full-size with  $1240 \times 1110$  pixels, half size with  $690 \times 555$  pixels and the third resolution with  $413 \times 370$ . Middlebury 2014 [106] is an RGB-D datasets unlike the other version.

## 4 Evaluation and Discussion

Due to the unavailability of a large dataset of damaged painting images and the novelty of the image inpainting topic, researchers find it difficult to obtain datasets for training their methods [107]. For that, most researchers use existing datasets like USC-SIPI, Paris StreetView, Places, ImageNet and others, and damage a set of images from these datasets for training their models and algorithms. The available methods in literature generate their own image inpainting datasets by adding some artificial distortion including noise [20], text [24], scratch [30], objects (shapes) [93], masks [95,97].

The evaluation metrics for image inpainting algorithms differ according to the using technique. In order to evaluate the efficiency of the proposed methods, researchers use some evaluation measures including mean squared error (MSE), Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [108]. For example, Zeng et al. [30] used these evaluation parameters to demonstrate the obtained results of repairing scratch and text in the images. Whereas Mo et al. [45] used the same metrics for evaluating the experiments for text and noise where Duan et al. [23] used the same metrics for evaluating the proposed method of removing the added objects to the images. In addition to the metrics used for evaluation, the category of images in the used dataset for evaluation can be different from one method to another. For example, some methods use RGB images while others evaluate their methods in RGB-D images or historical images. For that reason, we attempt to summarize the obtained results regarding each category of images used and the type of damaging in the images. In addition, this summarization is for sequential-based methods, which use the common evaluation metrics as shown in Table 4. From the table we can observe that most of these methods are evaluated on grayscale images like in [16,18,19,24,25,43,45] with some type of distortion including text, Gaussian noise (named Random in the papers) or some type of objects. We can also find some methods that analyze the three types of distortion like in [19], whereas other processes just for two types (text and noise) as presented in [18,43,45]. Also from the table, we can detect that the proposed methods use some recognized images in computer vision like Lena and Barbara images, which are used for testing the effectiveness of several methods. Methods that propose their approaches for image inpainting on RGB images use the same distortion categories including text, noise, and objects [13,15,22,23,30]. In addition, some researchers proposed methods for scratch analysis, which is a process of restoring the old images or images with damaged by some lines like in [15] and [29]. As shown in the tables, state-of-the-art methods use different images from the internet or some datasets and this is because of the lack of datasets for image inpainting.

We summarize the results according to the evaluation metrics used in each papers. In addition, in some works SNR or SSIM are used like in [32] and = others did not illustrate any evaluation metrics like in [34]. For that, in this papers we attempt to represent the PSNR metric that is used in the majority of related works [37,38].

<sup>13</sup> <http://vision.middlebury.edu/stereo/data/>.

**Table 4** Summarization of sequential-based methods evaluations

Image type	Distortion type	Method	PSNR	Images source
RGB	Text	Zeng et al. [30]	42.74	Internet
		Ružić et al. [15]	28.20	Internet
	Random	Duan et al. [23]	27.40	Internet
		Muddala et al. [13]	28.52	Internet
	Objects	Ding et al. [22]	29.19	Berkeley segmentation
		Duan et al. [23]	31.94	Self-collected
		Zeng et al. [30]	36.98	Internet
		Ghorai et al. [36]	24.61 (avg)	Internet
		Borole et al. [38]	16.95 (avg)	Internet
	Scratch	Ružić et al. [15]	23.89	Internet
		Yao et al. [29]	24.31	Thangka
RGB-D	Text	Xeu et al. [20]	27.65	Middlebury stereo
	Random	Xeu et al. [20]	27.18	Middlebury stereo
Grayscale	Text	Fan et al. [24]	32.83	Internet
		Lu et al. [19]	40.35	Barbara
		Mo et al. [45]	35.90	Barbara
		Geo et al. [18]	23.79	Lena
		Sridevi et al. [43]	40.63	Lena
	Random	Jin et al. [16]	38.04 (50%)	Barbara
		Jin et al. [16]	38.09 (50%)	Lena
		Geo et al. [18]	25.71 (20%), 29.39 (40%)	Lena
		Lu et al. [19]	30.31 (25%), 28.61 (15%)	Barbara
		Mo et al. [45]	39.65 (25%)	Barbara
		Sridevi et al. [43]	37.51 (40%)	Lena (512x512)
	Objects	Zhang et al. [33]	37.512	barbara
		Geo et al. [18]	29.18	Lena
		Fan et al. [24]	22.63	Lena
		Jiang et al. [25]	31.8	Lena
		Lu et al. [19]	39.05	Lena
		Gao et al. [37]	38.897	Lena
			34.868	Aerial
			42.69	Truck

With deep learning techniques, any task in computer vision can be performed with automatic learning using different unsupervised features, unlike the sequential-based method. The learning is made using convolutional neural networks (CNNs) that makes several computer vision tasks more improved in terms of robustness and most simple in terms of features suitable for each task. For image inpainting methods that use CNN, as described in the previous section, the effectiveness of each approach is related to the size and type of the data used and the architecture implemented. The evaluation of these methods is the same as for sequential-based methods. PSNR (the distance at the pixel level) and SSIM (similarity between two images) are used for evaluating the robustness of repairing damaged images under different categories of distortion including scratch, text, noise or random region (Blocks) added to the



**Table 5** Performance of CNN-based methods

Method	Datasets	Distortion type	PSNR	SSIM	Resolution
Yan et al. [46]	Paris StreetView Places	Random region	26.51	0.9	256 × 256
Cai et al. [50]	Berkley segmentation	Horiz/verti lines	38.22	0.9885	33 × 33
		Text	36.21	0.9741	
		Text (Lena)	36.67	0.9714	
Sidorov and Hardeberg [52]	Indian Pines	Vertical lines	37.54	0.979	145 × 145
Zeng et al. [53]	Places	Region blocks	–	0.7809	256 × 256
Liu et al. [54]	Paris StreetView CelebA	Random region	32.67	0.972	256 × 256
		Random region	34.69	0.989	
Pathak et al. [55]	Paris Street	Random region	17.59	–	227 × 227
Hsu et al. [57]	ImageNet	Random region	25.43	0.69	256 × 256
Xiang et al. [59]	Internet images	Scratch (old images)	27.42	0.9481	256 × 256
Alilou and Yaghmaee [60]	Internet images	Lena (scratch)	39.64	–	256 × 256
		Lena (text)	36.38	–	
		Lena (Noise 50%)	29.59	–	
		Lena (Noise 70%)	24.67	–	
Liao et al. [61]	Paris StreetView ImageNet	Scratch	19.94	0.59	256 × 256
		region block	17.67	0.52	
Cai et al. [62]	Paris StreetView ImageNet	Random region	6.69	0.19	256 × 256
Hertz et al. [63]	Microsoft COCO val2014	Text and symbols	37.63	0.9889	256 × 256
Zhao et al. [48]	Lung CT Test	Blocks	33.86	0.9781	256 × 256
	ZUBAL CT		36.33	0.9838	
Su et al. [65]	Places365	Random area	28.436	0.902	256 × 256
Ma et al. [66]	Places2	Masks (10–20%)	29.52	0.942	256 × 256
		Masks (40–50%)	23.34	0.807	
		Masks (50–60%)	22.04	0.755	
Liu et al. [68]	Places2	Masks (10–20%)	30.25	0.959	256 × 256
		Masks (40–50%)	22.57	0.78	
Guo et al. [69]	CelebA	Masks (10–20%)	33.66	0.9820	256 × 256
		Masks (20–30%)	29.01	0.9548	

image. Table 5 represents CNN-based methods for image inpainting and their performance evaluation and the datasets used, the type of distortion, evaluation metrics and the resolution of the images used in training. It is obvious that the performance of such methods is related to the type of distortion. For example, images damaged by blocks are less accurate in terms of PSNR values. The algorithms [50,52,60,63] can handle the added visual motifs like text or lines with a good performance. In addition, the performance is influenced by the percentage of added noise to the images. For the new dataset used for image inpainting, including Paris StreetView, Places or ImageNet that contains a large scale of data which are also different types of images, the algorithm's accuracy can be less than the others approaches using another

dataset [46,55,61,62]. This change of accuracy is related to the diversity of the images in these datasets.

Some proposed method presents their obtained results with a description of the different parameters used in the training phase which facilitate the comparison process. for example, the proposed architectures in [66,68,69] use the same masks for damaging the images before training their models. the obtained results depend on the area damaged in the images using masks. All the methods succeed to inpaint the images by a good quality when the images in masked by 10–20%. when the mask covers more than 40% the performance decrease. For example in [66] the PSNR value become 22.04 for 50–60% while it was 29.52, 10–20%.

Each method either makes a visual or qualitative evaluation; or an evaluation using metrics or quantitative evaluation. The quantitative evaluation, using PSNR and SSIM metrics, is performed also for image inpainting with GAN-based methods. In some cases, these metrics do not mean that qualitative results are better. This is related to the ground truth that should be unique [71]. Also, some methods for image inpainting are better for a certain category of images as well as the type of distortion used. Table 6 shows a number of GAN-based methods for image processing with a description of the used datasets and the evaluation metrics used for each method. In [75] the evaluation is made using many metrics depending on the position of the damaged region (block) including center, left, right, up, and down. But here we choose to present just PSNR and SSSIM of the image inpainting results on the images where the block is located in the center. In [73], two datasets are used with two types of distortion including blocks and free-form masks, which are categories of scratch painted with bold lines. For this example, we can see that the inpainting of scratch is more accurate than repairing the blocks. This becomes obvious from the fact that the blocks can take a region of the images where the scratch can take distributed small regions in the image.

Also from Table 6, we can observe that all quoted methods can recover missing regions with some difference in the accuracy of each method in terms of PSNR and SSIM metrics. For example, the methods presented in [91] and [92] are very close in terms of obtained values of PSNR. Also for the method [73,76] and [88], the PSNR values on CelebA dataset are 25.6 and 25.56 respectively. The convergence of the results is caused by the use of the same techniques (GAN) with some differences in the models.

As mentioned above, the unavailability of the datasets for image inpainting make the comparison between these methods difficult. Also, each author uses different masks and type of distortion.

## 4.1 Computational Time

Computational time represents a challenge for many tasks in computer vision, especially for real-time applications. Also with the speed of development of deep learning methods (i.e. from CNN to GAN), the training time, training speed, inference time, becomes a concern for image/video processing methods. For the image inpainting which presents a new challenge in computer vision, the computational time, or the other term related to it, is not much analyzed by the authors in the state-of-the-art methods except some of them. The existing methods describe either the time of training, inference or the training speed for inpainting an image. By the following, each one of the methods that considers the concept of time is presented:

- *Running time* In [51] the authors compute the average running time of each tested image with a resolution of  $256 \times 256$ . Where the results are 2s for each image. in [54] the proposed architecture takes 0.82s per image because of the use of a CSA layer that increases the computational time.

**Table 6** GAN-based performance results

Method	Datasets	Distortion type	PSNR	SSIM	Resolution
Chen and Hu [71]	CelebA	Center block	21.45	0.851	64 × 64
	Stanford Cars		15.02	0.725	
Shin et al. [76], Sagong et al. [73]	CelebA	Blocks	25.6	0.901	256 × 256
		Free-form mask	28.6	0.929	
	Places	Blocks	21.5	0.839	
		Free-form mask	25.2	0.889	
Wang et al. [75]	Cityscapes	Region blocks	16.06	0.4820	256 × 256
Victoria et al. [80]	CelebA	Random region	23.06	0.9341	64 × 64
	SVHN		26.23	0.8969	
	SceneNet dataset	Masks (RGB-D)	22.35	0.891	256 × 256
Nazeri et al. [87]	Places	Masks (10–20%)	27.95	0.920	256 × 256
		Masks (40–50%)	21.16	0.731	
Li et al. [88]	CelebA	Blocks	25.56	0.953	256 × 256
	Places2	Blocks	21.02	0.864	
	Paris StreetView	Blocks	24.33	0.867	
Li et al. [89]	Places2	Blocks	22.32	0.846	256 × 256
Yuan et al. [92]	CelebA	Blocks (center)	22.099	0.967	256 × 256
Yeh et al. [91]	CelebA	Blocks (center)	21.3	–	256 × 256

- *Training speed* The training speed is the evaluation time metric presented in [49] for describing the computational cost of the proposed architecture for training which was 7 fps (frame per second).
- *Inference time* The inference time is presented in [47] for each inpainting the RGB-D images using different Dept sensor including ORB-SLAM and Kinect depth map, and LIDAR depth map. For ORB-SLAM and Kinect depth map the inference time is about 30 ms where the resolution of the images is  $147 \times 109$  and about 200 ms for the full image resolution of  $640 \times 480$ . For the LIDAR depth map, the inference time is about 100 ms at  $608 \times 160$  image resolution. In the same context, the inference time takes 38 s to complete the image inpainting in [71].
- *Training time* In some works, the authors declare the time needed for training their model. for example, in [71] the training process costs 169 m for CelebA dataset and 66 m for Stanford Cars dataset where the architecture is implemented on an NVIDIA GTX 1080Ti GPU. The same architecture is trained on a CPU (i5-7400, 3.00GHz) and the process takes about 42 h.

## 5 Conclusions

Image inpainting is an important task for computer vision applications, due to large modified data using image editing tools. From these applications, we can find wireless image coding and transmission, image quality enhancement, image restoration and others. In this paper, a brief image inpainting review is presented. Different categories of approaches have been presented including sequential-based (approaches without learning), CNN-based approach and GAN-based approaches. We also attempt to collect the approaches that handle different

types of distortion in images such as text, objects added, scratch, and noise as well as several categories of data like RGB, RGB-D, historical images. A good alternative to these conventional features is the learned ones, e.g. deep learning, which has more generalization ability in more complicated scenarios. To be effective, these models need to be trained on a large amount of data. For that, we summarized the used datasets for training these models. In order to summarize the different analyzed cases and their performance, tabulated the evaluation performing the types of data, the datasets and the metrics used for each approach for each category of methods.

To conclude, there is no method that can inpaint all types of distortion in images, but using learning techniques provides some promising results for each category of the analyzed cases.

**Acknowledgements** This publication was made by NPRP grant # NPRP8-140-2-065 from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

## References

1. Yu J, Tan M, Zhang H, Tao D, Rui Y (2019) Hierarchical deep click feature prediction for fine-grained image recognition. *IEEE Trans Pattern Anal Mach Intell*. <https://doi.org/10.1109/TPAMI.2019.2932058>
2. Yu J, Zhu C, Zhang J, Huang Q, Tao D (2019) Spatial pyramid-enhanced NetVLAD with weighted triplet loss for place recognition. *IEEE Trans Neural Netw Learn Syst*. <https://doi.org/10.1109/TNNLS.2019.2908982>
3. Zhang J, Yu J, Tao D (2018) Local deep-feature alignment for unsupervised dimension reduction. *IEEE Trans Image Process* 27(5):2420–2432
4. Yu J, Tao D, Wang M, Rui Y (2014) Learning to rank using user clicks and visual features for image retrieval. *IEEE Trans Cybern* 45(4):767–779
5. Yu J, Yang X, Gao F, Tao D (2016) Deep multimodal distance metric learning using click constraints for image ranking. *IEEE Trans Cybern* 47(12):4014–4024
6. Hong C, Yu J, Wan J, Tao D, Wang M (2015) Multimodal deep autoencoder for human pose recovery. *IEEE Trans Image Process* 24(12):5659–5670
7. Hong C, Yu J, Tao D, Wang M (2014) Image-based three-dimensional human pose recovery by multiview locality-sensitive sparse retrieval. *IEEE Trans Ind Electron* 62(6):3742–3751
8. Yu J, Zhang B, Kuang Z, Lin D, Fan J (2016) iPrivacy: image privacy protection by identifying sensitive objects via deep multi-task learning. *IEEE Trans Inf Forensics Secur* 12(5):1005–1016
9. Hong C, Yu J, Zhang J, Jin X, Lee K-H (2018) Multi-modal face pose estimation with multi-task manifold deep learning. *IEEE Trans Ind Inform* 15(7):3952–3961
10. Elharrouss O, Abbad A, Moujahid D, Riffi J, Tairi H (2016) A block-based background model for moving object detection. *Electron Lett Comput Vis Image Anal* 15(3):0017–31
11. Abbad A, Elharrouss O, Abbad K, Tairi H (2018) Application of meemd in post-processing of dimensionality reduction methods for face recognition. *IET Biom* 8(1):59–68
12. Moujahid D, Elharrouss O, Tairi H (2018) Visual object tracking via the local soft cosine similarity. *Pattern Recognit Lett* 110:79–85
13. Muddala SM, Olsson R, Sjöström M (2016) Spatio-temporal consistent depth-image-based rendering using layered depth image and inpainting. *EURASIP J Image Video Process* 2016(1):9
14. Isogawa M, Mikami D, Iwai D, Kimata H, Sato K (2018) Mask optimization for image inpainting. *IEEE Access* 6:69728–69741
15. Ružić T, Pižurica A (2014) Context-aware patch-based image inpainting using markov random field modeling. *IEEE Trans Image Process* 24(1):444–456
16. Jin KH, Ye JC (2015) Annihilating filter-based low-rank hankel matrix approach for image inpainting. *IEEE Trans Image Process* 24(11):3498–3511
17. Kawai N, Sato T, Yokoya N (2015) Diminished reality based on image inpainting considering background geometry. *IEEE Trans Vis Comput Gr* 22(3):1236–1247
18. Guo Q, Gao S, Zhang X, Yin Y, Zhang C (2017) Patch-based image inpainting via two-stage low rank approximation. *IEEE Trans Vis Comput Gr* 24(6):2023–2036
19. Lu H, Liu Q, Zhang M, Wang Y, Deng X (2018) Gradient-based low rank method and its application in image inpainting. *Multimed Tools Appl* 77(5):5969–5993

20. Xue H, Zhang S, Cai D (2017) Depth image inpainting: improving low rank matrix completion with low gradient regularization. *IEEE Trans Image Process* 26(9):4311–4320
21. Liu J, Yang S, Fang Y, Guo Z (2018) Structure-guided image inpainting using homography transformation. *IEEE Trans Multimed* 20(12):3252–3265
22. Ding D, Ram S, Rodríguez JJ (2018) Image inpainting using nonlocal texture matching and nonlinear filtering. *IEEE Trans Image Process* 28(4):1705–1719
23. Duan J, Pan Z, Zhang B, Liu W, Tai X-C (2015) Fast algorithm for color texture image inpainting using the non-local CTV model. *J Glob Optim* 62(4):853–876
24. Fan Q, Zhang L (2018) A novel patch matching algorithm for exemplar-based image inpainting. *Multimed Tools Appl* 77(9):10807–10821
25. Jiang W (2016) Rate-distortion optimized image compression based on image inpainting. *Multimed Tools Appl* 75(2):919–933
26. Alilou VK, Yaghmaee F (2017) Exemplar-based image inpainting using svd-based approximation matrix and multi-scale analysis. *Multimed Tools Appl* 76(5):7213–7234
27. Wang W, Jia Y (2017) Damaged region filling and evaluation by symmetrical exemplar-based image inpainting for thangka. *EURASIP J Image Video Process* 2017(1):38
28. Wei Y, Liu S (2016) Domain-based structure-aware image inpainting. *Signal Image Video Process* 10(5):911–919
29. Yao F (2018) Damaged region filling by improved criminisi image inpainting algorithm for thangka. *Clust Comput* 22:1–9
30. Zeng J, Fu X, Leng L, Wang C (2019) Image inpainting algorithm based on saliency map and gray entropy. *Arabian J Sci Eng* 44(4):3549–3558
31. Zhang D, Liang Z, Yang G, Li Q, Li L, Sun X (2018) A robust forgery detection algorithm for object removal by exemplar-based image inpainting. *Multimed Tools Appl* 77(10):11823–11842
32. Wali S, Zhang H, Chang H, Wu C (2019) A new adaptive boosting total generalized variation (TGV) technique for image denoising and inpainting. *J Vis Commun Image Represent* 59:39–51
33. Zhang Q, Lin J (2012) Exemplar-based image inpainting using color distribution analysis. *J Inf Sci Eng* 28(4):641–654
34. Liu Y, Caselles V (2012) Exemplar-based image inpainting using multiscale graph cuts. *IEEE Trans Image process* 22(5):1699–1711
35. Qin C, Chang C-C, Chiu Y-P (2013) A novel joint data-hiding and compression scheme based on SMVQ and image inpainting. *IEEE Trans Image Process* 23(3):969–978
36. Ghorai M, Samanta S, Mandal S, Chanda B (2019) Multiple pyramids based image inpainting using local patch statistics and steering kernel feature. *IEEE Trans Image Process* 28(11):5495–5509
37. Gao J, Zhu J, Nie K, Xu J An image inpainting method for interleaved 3D stacked image sensor, *IEEE Sensors Journal*
38. Borole RP, Bonde SV (2007) Image restoration using prioritized exemplar inpainting with automatic patch optimization. *J Inst Eng India Ser B* 98(3):311–319
39. Hoeltgen L (2017) Understanding image inpainting with the help of the Helmholtz equation. *Math Sci* 11(1):73–77
40. Zhang T, Gelman A, Laronga R (2017) Structure-and texture-based fullbore image reconstruction. *Math Geosci* 49(2):195–215
41. Li H, Luo W, Huang J (2017) Localization of diffusion-based inpainting in digital images. *IEEE Trans Inf Forensics Secur* 12(12):3050–3064
42. Li K, Wei Y, Yang Z, Wei W (2016) Image inpainting algorithm based on TV model and evolutionary algorithm. *Soft Comput* 20(3):885–893
43. Sridevi G, Kumar SS (2019) Image inpainting based on fractional-order nonlinear diffusion for image reconstruction. *Circuit Syst Signal Process* 38(8):1–16
44. Jin X, Su Y, Zou L, Wang Y, Jing P, Wang ZJ (2018) Sparsity-based image inpainting detection via canonical correlation analysis with low-rank constraints. *IEEE Access* 6:49967–49978
45. Mo J, Zhou Y (2018) The research of image inpainting algorithm using self-adaptive group structure and sparse representation. *Cluster Comput* 22(1):1–9
46. Yan Z, Li X, Li M, Zuo W, Shan S (2018) Shift-net: Image inpainting via deep feature rearrangement. In: *Proceedings of the European conference on computer vision (ECCV)*, pp 1–17
47. Weerasekera CS, Dharmasiri T, Garg R, Drummond T, Reid I (2018) Just-in-time reconstruction: inpainting sparse maps using single view depth predictors as priors. In: *2018 IEEE international conference on robotics and automation (ICRA)*, IEEE, pp 1–9
48. Zhao J, Chen Z, Zhang L, Jin X (2018) Unsupervised learnable sinogram inpainting network (sin) for limited angle ct reconstruction. *arXiv preprint [arXiv:1811.03911](https://arxiv.org/abs/1811.03911)*

49. Chang Y-L, Yu Liu Z, Hsu W (2019) VORNet: spatio-temporally consistent video inpainting for object removal. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 0–0
50. Cai N, Su Z, Lin Z, Wang H, Yang Z, Ling BW-K (2017) Blind inpainting using the fully convolutional neural network. *Vis Comput* 33(2):249–261
51. Zhu X, Qian Y, Zhao X, Sun B, Sun Y (2018) A deep learning approach to patch-based image inpainting for forensics. *Signal Process Image Commun* 67:90–99
52. Sidorov O, Hardeberg JY (2019) Deep hyperspectral prior: denoising, inpainting, super-resolution. *arXiv preprint arXiv:1902.00301*
53. Zeng Y, Fu J, Chao H, Guo B (2019) Learning pyramid-context encoder network for high-quality image inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1486–1494
54. Liu H, Jiang B, Xiao Y, Yang C (2019) Coherent semantic attention for image inpainting. *arXiv preprint arXiv:1905.12384*
55. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA (2016) Context encoders: feature learning by inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2536–2544
56. Sasaki K, Iizuka S, Simo-Serra E, Ishikawa H (2017) Joint gap detection and inpainting of line drawings. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5725–5733
57. Hsu C, Chen F, Wang G (2017) High-resolution image inpainting through multiple deep networks. In: 2017 international conference on vision, image and signal processing (ICVISP), IEEE, pp 76–81
58. Nakamura T, Zhu A, Yanai K, Uchida S (2017) Scene text eraser. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR), IEEE, Vol 1 pp 832–837
59. Xiang P, Wang L, Cheng J, Zhang B, Wu J (2017) A deep network architecture for image inpainting. In: 2017 3rd IEEE international conference on computer and communications (ICCC), IEEE, pp 1851–1856
60. Alilou VK, Yaghmaee F (2015) Application of GRNN neural network in non-texture image inpainting and restoration. *Pattern Recognit Lett* 62:24–31
61. Liao L, Hu R, Xiao J, Wang Z (2019) Artist-net: decorating the inferred content with unified style for image inpainting. *IEEE Access* 7:36921–36933
62. Cai X, Song B (2018) Semantic object removal with convolutional neural network feature-based inpainting approach. *Multimed Syst* 24(5):597–609
63. Hertz A, Fogel S, Hanocka R, Giryas R, Cohen-Or D (2019) Blind visual motif removal from a single image. *arXiv preprint arXiv:1904.02756*
64. Zhao Y, Price B, Cohen S, Gurari D (2019) Guided image inpainting: replacing an image region by pulling content from another image. In: 2019 IEEE winter conference on applications of computer vision (WACV), IEEE, pp 1514–1523
65. Su Y-Z, Liu T-J, Liu K-H, Liu H-H, Pei S-C (2019) Image inpainting for random areas using dense context features. In: 2019 IEEE international conference on image processing (ICIP), IEEE, pp 4679–4683
66. Ma Y, Liu X, Bai S, Wang L, He D, Liu A (2019) Coarse-to-fine image inpainting via region-wise convolutions and non-local correlation. In: Proceedings of the 28th international joint conference on artificial intelligence, AAAI press, pp 3123–3129
67. Ke J, Deng J, Lu Y (2019) Noise reduction with image inpainting: an application in clinical data diagnosis. In: ACM SIGGRAPH 2019 posters, ACM, p 88
68. Liu S, Guo Z, Chen J, Yu T, Chen Z (2019) Interleaved zooming network for image inpainting. In: 2019 IEEE international conference on multimedia & expo workshops (ICMEW), IEEE, pp 673–678
69. Guo Z, Chen Z, Yu T, Chen J, Liu S (2019) Progressive image inpainting with full-resolution residual network. In: Proceedings of the 27th ACM international conference on multimedia, ACM, pp 2496–2504
70. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
71. Chen Y, Hu H (2019) An improved method for semantic image inpainting with gans: progressive inpainting. *Neural Process Lett* 49(3):1355–1367
72. Li J, Song G, Zhang M (2008) Occluded offline handwritten Chinese character recognition using deep convolutional generative adversarial network and improved GoogLeNet. *Neural Comput Appl* 1–15
73. Shin YG, Sagong MC, Yeo YJ, Kim SW, Ko SJ (2019) Pepsi++: fast and lightweight network for image inpainting. *arXiv preprint arXiv:1905.09010*
74. Wang H, Jiao L, Wu H, Bie R (2019) New inpainting algorithm based on simplified context encoders and multi-scale adversarial network. *Procedia Comput Sci* 147:254–263
75. Wang C, Xu C, Wang C, Tao D (2018) Perceptual adversarial networks for image-to-image transformation. *IEEE Trans Image Process* 27(8):4066–4079



76. Sagong M-c, Shin Y-g, Kim S-w, Park S, Ko S-j (2019) Pepsi: fast image inpainting with parallel decoding network, In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 11360–11368
77. Dharmo H, Tateno K, Laina I, Navab N, Tombari F (2019) Peeking behind objects: layered depth prediction from a single image. *Pattern Recognit Lett* 125:333–340
78. Elharrouss O, Al-Maadeed N, Al-Maadeed S (2019) Video summarization based on motion detection for surveillance systems. In: 15th international wireless communications & mobile computing conference (IWCMC). IEEE, pp 366–371
79. Almaadeed N, Elharrouss O, Al-Maadeed S, Bouridane A, Beghdadi A (2019) A novel approach for robust multi human action detection and recognition based on 3-dimensional convolutional neural networks. *arXiv preprint arXiv:1907.11272*
80. Vitoria P, Sintés J, Ballester C (2018) Semantic image inpainting through improved Wasserstein generative adversarial networks. *arXiv preprint arXiv:1812.01071*
81. Dong J, Yin R, Sun X, Li Q, Yang Y, Qin X (2018) Inpainting of remote sensing sst images with deep convolutional generative adversarial network. *IEEE Geosci Remote Sens Lett* 16(2):173–177
82. Lou S, Fan Q, Chen F, Wang C, Li J, (2018) Preliminary investigation on single remote sensing image inpainting through a modified gan, In: 2018 10th IAPR workshop on pattern recognition in remote sensing (PRRS). IEEE 1–6
83. Salem N.M, Mahdi HM, Abbas H (2018) Semantic image inpainting vsing self-learning encoder-decoder and adversarial loss, In: 2018 13th international conference on computer engineering and systems (ICCES), IEEE, pp 103–108
84. Liu H, Lu G, Bi X, Yan J, Wang W (2018) Image inpainting based on generative adversarial networks, In: 2018 14th international conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD), IEEE, pp 373–378
85. Han X, Wu Z, Huang W, Scott MR, Davis LS (2019) Compatible and diverse fashion image inpainting. *arXiv preprint arXiv:1902.01096*
86. Jiao L, Wu H, Wang H, Bie R (2019) Multi-scale semantic image inpainting with residual learning and GAN. *Neurocomputing* 331:199–212
87. Nazeri K, Ng E, Joseph T, Qureshi F, Ebrahimi M (2019) Edgeconnect: generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212*
88. Li A, Qi J, Zhang R, Ma X, Ramamohanarao K (2019) Generative image inpainting with submanifold alignment. *arXiv preprint arXiv:1908.00211*
89. Li A, Qi J, Zhang R, Kotagiri R (2019) Boosted gan with semantically interpretable information for image inpainting, In: 2019 international joint conference on neural networks (IJCNN), IEEE, pp 1–8
90. Armanious K, Mecky Y, Gatidis S, Yang B (2019) Adversarial inpainting of medical image modalities, In: ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP) IEEE, pp 3267–3271
91. Yeh R A, Chen C, Yian Lim T, Schwing AG, Hasegawa-Johnson M, Do MN (2017) Semantic image inpainting with deep generative models, In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5485–5493
92. Yuan L, Ruan C, Hu H, Chen D (2019) Image inpainting based on patch-GANs. *IEEE Access* 7:46411–46421
93. Doersch C, Singh S, Gupta A, Sivic J, Efros AA (2015) What makes paris look like paris? *Commun ACM* 58(12):103–110
94. Zhou B, Lapedriza A, Khosla A, Oliva A, Torralba A (2017) Places: a 10 million image database for scene recognition. *IEEE Trans Pattern Anal Mach Intell* 40(6):1452–1464
95. Xiong W, Yu J, Lin Z, Yang J, Lu X, Barnes C, Luo J (2019) Foreground-aware image inpainting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5840–5848
96. Martin D, Fowlkes C, Tal D, Malik J, et al (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, *Iccv Vancouver*
97. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al (2015) Imagenet large scale visual recognition challenge. *Int J Comput vis* 115(3):211–252
98. Liu Z, Luo P, Wang X, Tang X (2018) Large-scale celebfaces attributes (celeba) dataset, Retrieved August 15
99. Baumgardner M F, Biehl LL, Landgrebe DA (2015) 220 band aviris hyperspectral image data set: June 12, 1992 Indian pine test site 3. Purdue University Research Repository 10: R7RX991C
100. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: common objects in context. In: European conference on computer vision. Springer, Cham, pp 740–755

101. Karatzas D, Shafait F, Uchida S, Iwamura M, i Bigorda LG, Mestre SR, Mas J, Mota DF, Almazan JA, De Las Heras LP (2013) ICDAR 2013 robust reading competition. In: 2013 12th international conference on document analysis and recognition, IEEE, pp 1484–1493
102. McCormac J, Handa A, Leutenegger S, Davison AJ (2016) Scenenet RGB-D: 5m photorealistic images of synthetic indoor trajectories with ground truth. arXiv preprint [arXiv:1612.05079](https://arxiv.org/abs/1612.05079)
103. Krause J, Stark M, Deng J, Fei-Fei L (2013) 3D object representations for fine-grained categorization, In: Proceedings of the IEEE international conference on computer vision workshops, pp 554–561
104. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3213–3223
105. Hirschmuller H, Scharstein D (2007) Evaluation of cost functions for stereo matching, In: 2007 IEEE conference on computer vision and pattern recognition, IEEE, pp 1–8
106. Scharstein D, Hirschmüller H, Kitajima Y, Krathwohl G, Nešić N, Wang X, Westling P (2014) High-resolution stereo datasets with subpixel-accurate ground truth. In: German conference on pattern recognition. Springer, cham, pp 31–42
107. Jboor NH, Belhi A, Al-Ali AK, Bouras A, Jaoua A (2019) Towards an inpainting framework for visual cultural heritage. In: 2019 IEEE Jordan international joint conference on electrical engineering and information technology (JEEIT), IEEE, pp 602–607
108. Elharrouss O, Abbad A, Moujahid D, Tairi H (2017) Moving object detection zone using a block-based background model. IET Comput Vis 12(1):86–94

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval, sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

[onlineservice@springernature.com](mailto:onlineservice@springernature.com)