IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 6, 2020, accepted May 20, 2020, date of publication June 1, 2020, date of current version June 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2999076

Single Image Haze Removal Using Deep Cellular Automata Learning

SURASAK TANGSAKUL AND SARTRA WONGTHANAVASU, (Member, IEEE)
Department of Computer Science, Faculty of Science, Khon Kaen University, Khon Kaen 40002, Thailand

Corresponding author: Sartra Wongthanavasu (Wongsar@kku.ac.th)

This work was supported in part by the Thailand Research Fund (TRF) under Grant RTA6080013, and in part by the Computer Science Department, Khon Kaen University, Thailand.

ABSTRACT Deep learning is one of the most popular approaches to machine learning, which has been widely used for classification. In this paper, we propose a novel learning method based on a combination of an idea of the deep learning approach and the cellular automata model, called DeepCA for single image haze removal. DeepCA's learning is divided into two main parts. The first part is a cellular automata-based deep feature extraction: multi-layer cellular automata with the rules are used to extract the data feature matrices of the image, in which the matrices can be divided into several layers. Then, the score matrices were generated as the model in which was trained by the cellular automata rules. The second part is a decision stage: we used the score matrices to the mapping between the proper data. For demonstration, we take the single image haze removal task as an example to confirm the capability of the proposed method. In this regard, the dichromatic model is chosen as the major model to remove the haze of the image. The multi-layer cellular automata with the rules work as a mechanical extractor of the light source feature of the hazy image. The decision stage of DeepCA performs as the recognizer for properly predicting the global light source for dehazing. This aims to improve the light source and the transmission map that they are important compositions for haze-free image restoration. For performance evaluation, we perform quantitative and qualitative measures. For the qualitative performance of the haze removal, DeepCA did not even cause the halo artifact effect that occurred in other haze removal algorithms. The empirical results in quantitative measures show that DeepCA improved intensity, color saturation quality, and halo artifact when compared with the state-of-the-art methods.

INDEX TERMS Cellular automata, deep learning, single image, dehazing.

I. INTRODUCTION

Nowadays haze removal algorithms (or dehazing algorithms) are still challenging research problems in the field of image processing due to weather and environment changes. Several haze removal algorithms were published continuously due to their fruitful applications. The haze removal algorithm is not only the most important process for landscape or outdoor photographing task, but also significantly improves the performance of computer vision applications, such as in the preprocessing of image segmentation, object detection or image classification. Actually, there are two types of small particles diffused in the air: haze and fog that are due to different in natural processes. Haze is constituted of aerosol, which is a dispersed system of small dust particles suspended in gas, such as combustion products or volcanic ashes. On the other hand, fog evolves when the relative humidity of an air parcel reaches saturation, then some of the nuclei grow by condensation into liquid droplets [1], [2]. In addition, haze particles are larger than air molecules but smaller than fog droplets [2]. However, in hazy images, we can observe the effect of haze or fog that even have low contrast, faint color, and shifted luminance.

In the past decades, researchers used various techniques to dehazing and enhancing the image contrast, color saturation, and restored the important details of the image. Based on our investigation of image dehazing researches, we found that there are two approaches of haze removal methods consisting of multiple and single images-based haze removal. Multiple image-based haze removal methods usually required multiple images to perform dehazing. For example, polarization-based methods restore the scene depth information from

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Olague.

different degrees of polarization property of multiple images [3], [4]. Similarly, [5] and [6] capture the multiple images of the same scene under different weather conditions to be used as reference images with the clear weather condition. However, these methods with multiple reference images have a limitation in online image dehazing applications and may need a special imaging sensor [3], [6]. Alternatively, the single image-based haze removal method relies on the typical characteristics of a haze-free image. Recently, researchers used various techniques for single image-based haze removal. For example, contrast enhancement algorithms were proposed by [7]–[9], Multi-scale fusion algorithms were proposed by [10] and [11], Retinex dehazing algorithm was proposed by [12], and the most popular conventional single haze removal algorithm is based on physics-based algorithm or dichromatic model [2], [13]–[19].

The dichromatic model consists of equation parts suitable for the hazy image related to computer vision theory that is easy to understand. This model consists of four significant steps: atmospheric light estimation, transmission map estimation, transmission map refinement, and haze-free image reconstruction. The details will be mentioned in the next section. However, the dichromatic model still has some limitations. Tan [14] observed the difference between the contrast of the hazy image and haze-free image. He proposed a method that takes into account the characteristic that a haze-free image has higher contrast than a hazy image. By maximizing the local contrast value of an image, it enhances the visibility but introduces blocking artifacts around depth discontinuities. Fattal [15] proposed a method that estimated the albedo of the scene by the assumption of medium transmission that the transmission and surface shading are locally incorrect, especially under a dense haze area of the image. He *et al.* [13] proposed a novel dark channel prior (DCP) by observing the property of haze-free outdoor images. The DCP is based on the property of the dark pixels (in the dark channel), which have a very low intensity in the color channel except for the sky region. Owing to its effectiveness in dehazing, the majority of recent dehazing techniques [14]–[20] adopted the DCP in their works. Nevertheless, the results of some recovered scenes are over-saturated in the sky region for the small patch size, and also contain halo artifacts or border effect for the large patch size. In this regard, these problems are solved by soft matting interpolation in the post-process of transmission map construction [13], bilateral filter [17], guided filter [21]. Zhu *et al.* [22] copes with the drawback of the sky region of the DCP method by proposed a fusion of luminance and dark channel prior (F-LDCP) method to effectively restore of long-shot haze-free images, especially for the sky region.

In this paper, an improved single image haze removal method is proposed based on deep learning approach and cellular automata theory called Deep Cellular Automata learning (DeepCA). The main contributions of this paper are summarized as follows.

1) We propose a novel deep learning method called DeepCA that used multi-layer cellular automata and the rules vector as the major of deep learning mechanism.

2) We propose a novel and efficient of single image haze removal algorithm based-on DeepCA that is dealing with the transmission map, small haze preserve parameter (ω), and atmospheric light ratio (ρ). DeepCA directly learns to the mapping between the hazy image and their transmission map. This is achieved by special design of DeepCA architecture and rules training algorithm. Moreover, the proposed parameters help DeepCA to preserve the natural appearance of images capable of producing more natural haze-free image without oversaturation problem and halo artifacts.

The remainder of the paper is organized as follows. In Section II, we present the problem statement and the dichromatic model for dehazing, Section III offers background of the hazy image and cellular automata model. In Section IV, we provide details of the proposed DeepCA in the case of single haze removal problem. Performance evaluation and experimental results are given in Section V, and Section VI provides a conclusion.

II. PROBLEM STATEMENT

A. HAZY IMAGE IN COMPUTER VISION

In computer vision and computer graphics, the traditional physical atmospheric scattering model or also known as the dichromatic model [2] has been widely used to describe the formation of hazy image and to dehazing [9]–[11], [13]–[22]. It can be defined in Eq.(1) as follow:

$$I(x, y) = I'(x, y)t(x, y) + A(1 - t(x, y)) \tag{1}$$

where I is the input hazy image, x and y are image coordinates, I' is the haze-free image or scene radiance, t is a transmission value, and A is the global atmospheric light in image space. Eq. (1) consists of two terms: the first term $I'(x, y)t(x, y)$ is direct transmission of haze-free image, and the second term $A(1-t(x, y))$ is called air light. The equation describes the relation of haze-free image, direct transmission light reflected value and the global atmospheric light. In this regard, [13] replaced $t(x, y)$ by $e^{-\beta d(x,y)}$ shown in the Eq. (2), where β is scattering coefficient of atmospheric light and $d(x, y)$ is desired scene depth.

$$t(x, y) = e^{-\beta d(x,y)} \tag{2}$$

Then, replace Eq. (2) in Eq. (1), resulting in Eq. (3).:

$$I(x, y) = I'(x, y)e^{-\beta d(x,y)} + A(1 - e^{-\beta d(x,y)}) \tag{3}$$

Eq. (3) describes the weather effect in hazy image, for $\beta \approx 0$ in clear weather we have $I(x, y) \approx I'(x, y)$. More recently, various algorithms of single image haze removal used the dichromatic model to dehazing. They have been recovering $I'(x, y)$ from the Eq.(1), that can rewrite as follow:

$$I'(x, y) = \frac{I(x, y) - A}{t(x, y)} + A \tag{4}$$

where I' is a haze-free image, I is a hazy image, x and y are image coordinates, t is a transmission value of the transmission map, and A is the global atmospheric light in the image.

For the single image haze removal algorithm based on the dichromatic model with dark channel prior, the processes of dehazing are as follows.

Firstly, the dark channel must be constructed from the hazy image before estimating the atmospheric light. From an empirical investigation on a number of the outdoor scene of the haze-free images, He *et al.* [13] observed that at least one color channel with some pixels in the image patch has very low intensity closed to zero. Thus, the value of the dark pixel in any x, y position ($I^{dark}(x, y)$) can be estimated from the three RGB channels by Eq. (5).

$$I^{dark}(x, y) = \min_{x,y\in\Omega(x,y)}(\min_{c\in\{r,g,b\}} I^c(x, y)) \tag{5}$$

where I^c is a color intensity of the pixel in red(r), green(g) and blue(b) color channel, and $\Omega(x, y)$ is any local patch of the pixel at the x, y position. This equation describes the minimum of color intensity from all three RGB channels in all patches for the dark channel. However, for different sizes of the local patch based on Eq. (5), the results of some recovered scenes are often low intensity, low brightness, and over-saturated image for small patch size and contain halo artifacts or border effect for large patch size.

Secondly, the atmospheric light is the brightest pixel in the image estimated from the dark channel. It is the most important parameter to restore the haze-free image because it affects all pixels of the restored image. Several DCP-based algorithms estimated the atmosphere light by Eq. (6). However, the researchers founded incorrect results when the image scene contains a large brightest area or bright objects, such as landscape images with the sky scene. Hence, they improved the Eq. (6) by defining a percentage of the top value of the brightest pixel (p) as the atmosphere light. For example, [13], and [20] used $p = 0.1$ of the brightest pixel in the dark channel to estimate the atmospheric light, [17] used $p = 0.2$ and [18] used the top 5 percentage and edge information to estimate the atmospheric light.

$$A = I(argmax_{(x,y)}(I^{dark}(x, y))) \tag{6}$$

where I is a hazy image, $argmax_{(x,y)}$ is the max value of the brightest pixel in the dark channel I^{dark}. The improved version of the atmospheric light estimation with parameter p is defined as follow:

$$A = I(p * argmax_{(x,y)}(I^{dark}(x, y))) \tag{7}$$

Thirdly, the transmission map I^{tran} is an image composition that maintained the image intensity information of the haze-free image. It can be obtained by using Eq. (8). This equation is constructed from the positive one subtracted by the dark channel of hazy image I^{dark} that I^c is normalized by atmospheric light A^c. The equation can be removed the haze thoroughly, but the image may seem unnatural and lose the feeling of image depth, this called under-estimation problem.

$$I^{tran}(x, y) = 1 - \min_{x,y\in\Omega(x,y)}(\min_{c\in\{r,g,b\}} \frac{I^c(x, y)}{A^c}) \tag{8}$$

To cope with the under-estimation problem, researchers proposed an idea to keep a small amount of haze for the distant objects in the image. He *et al.* [13] used a positive constant value $\omega(0 < \omega < 1)$ to retain the small number of haze; it is defined as the Eq. (9) as follow:

$$I^{tran}(x, y) = 1 - \omega \min_{x,y\in\Omega(x,y)}(\min_{c\in\{r,g,b\}} \frac{I^c(x, y)}{A^c}) \tag{9}$$

Xu *et al.* [20] added a positive constant value $\rho \in [0.08, 0.25]$ to cope with the same problem defined in Eq. (10):

$$I^{tran}(x, y) = 1 - \min_{x,y\in\Omega(x,y)}(\min_{c\in\{r,g,b\}} \frac{I^c(x, y)}{A^c}) + \rho \tag{10}$$

In general, the parameter ω and ρ are positive constant value in a range of $0 < \omega, \rho < 1$, Fig. 1 shows the differences between ω and ρ in the normal range (0 to 1) and over the range (> 1) that significantly affected the haze-free image. Thus, these parameters in the normal range are able to enhance the haze-free image in terms of haze retaining (see Fig. 1 (a) to (d)) and atmospheric light (or global light) tuning (see Fig. 1 (f) to (i)).

Finally, haze-free image restoration, the haze-free image is restored from the transmission map and atmosphere light information by the traditional equation defined in Eq. (4).

Regarding the problems above, there are two significant factors in hazy images that necessary to solve. The first one is the over-saturated results introduced by the atmosphere light estimation problem in the case of the images comprise the large brightest area. The second one is the unnatural results presented by the inappropriate transmission map. To address these problems, we propose a method for estimating the proper transmission map that avoids the atmosphere light estimation problem of the large brightest area and under-estimation problem using a novel deep cellular automata learning to identify the significant parameters. Besides, the proposed method will be completely restoring the haze-free image without any post-processing.

III. RELATED WORK

Many single image dehazing methods have been proposed in the literature. In this section, we investigate two majors categories, i.e., prior-based dehazing methods and learning-based dehazing methods. Then the basis of cellular automata and their application will be mentioned, which provides background knowledge to understand the design of DeepCA.

A. PRIOR-BASED DEHAZING

In past years, several single image dehazing algorithms rely on the traditional dichromatic model and prior-based method. For example, Tan [14] proposed a method that takes into account the characteristic that the haze-free image has higher contrast than the hazy image. The patch-based local contrast maximizing method is applied to enhance visibility results,

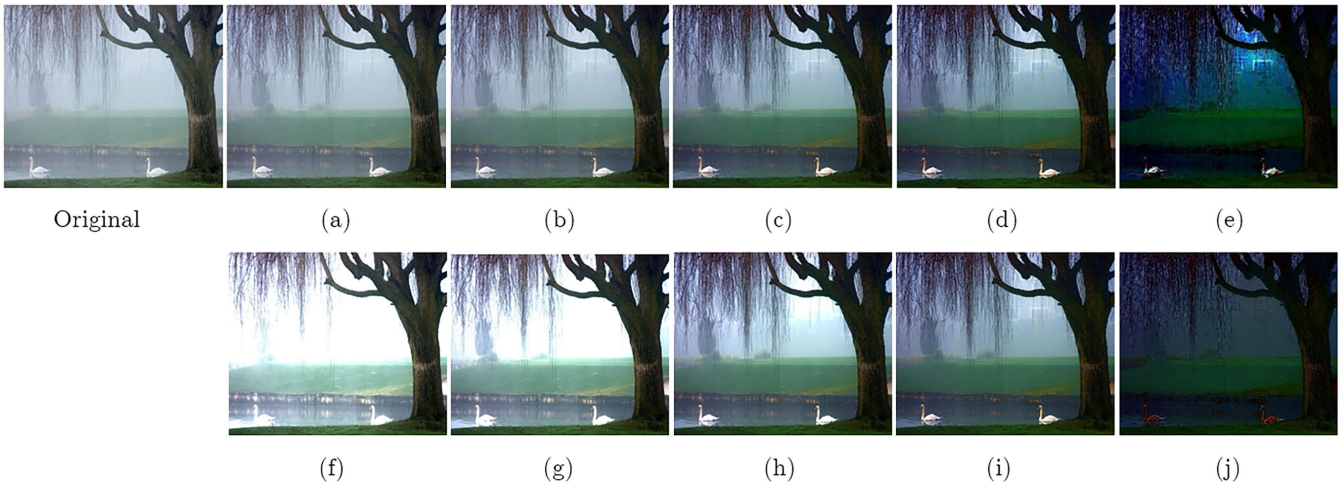

FIGURE 1. Effect of different ω value and atmospheric light value: (a)-(e) fixed atmospheric light value, $\omega = 0.5, 0.7, 0.9, 1.0$, and 1.2, respectively., (f)-(j) fixed $\omega = 0.95$, atmospheric light value divide by $\rho = 0.5, 0.7, 0.9, 1.0$, and 1.2, respectively.

but it is still introduced blocking artifacts around depth discontinuities. Fattal [23] estimated the albedo of the scene by the assumption of medium transmission that the transmission and surface shading are locally incorrect. This method can slightly reduce the haze and requires time-consuming to process. Then, Fattal [15] proposed a color-line method that the scene transmission is recovered based on the color-lines inside small image patches. However, some image condition is not sufficient to guarantee a correct classification of patches, and the method cannot operate on monochromatic images. He *et al.* [13] proposed a method that addresses the dark channel prior (DCP) by observing the property of the dark pixels in outdoor haze-free images. The transmission map is estimated from the dark channel for dehazing. Besides, the majority of recent dehazing techniques [16]–[22] also applied this approach in their works. Although these methods have achieved outstanding haze-free image results, some of the dehazing results are still needed to be further improved. For example, the recovered image of the image scene that contains the large brightest area or the sky region is over-saturated and also contain halo artifacts.

B. LEARNING-BASED DEHAZING

Due to the rapid development of deep learning approaches in computer vision tasks [24]–[28], the deep learning-based methods have been applied to single image dehazing. For instance, Cai *et al.* [29] proposed a DehazeNet based on the classical convolutional neural network (CNN) and the atmospheric scattering model to the mapping between the hazy image and the transmission map. The architecture of DehazeNet includes four sequential operations, i.e., feature extraction, multi-scale mapping, local extremum, and non-linear regression, which is constructed by three convolution layers, a max-pooling, a Maxout unit, and a BReLU activation function. Due to a general costly to collect a vast amount of labeled data for training deep models [25], they used the haze-free images obtained from the Internet as the training dataset and randomly sample from them patches of size 16×16. Li *et al.* [30] proposed an AOD-Net to learn a mapping function based on CNN and a re-formulated atmospheric scattering model. The AOD-Net is trained on the synthesized hazy image and tested on both synthetic and real natural images. Ren *et al.* [31] proposed a fusion-based encoder-decoder network called Gated Fusion Network (GFN), by learning the confidence map to directly restore the haze-free image without estimating the transmission map and atmospheric light. The GFN is trained on NYU2 dataset [32] and adopts the synthetic method in [33] to synthesize the training data. Li *et al.* [34] proposed a flexible cascaded network based on CNN and the atmospheric scattering model for single image dehazing, which considers the medium transmission and global atmospheric light jointly by two task-driven subnetworks. The cascaded CNN includes three parts: the shared hidden layers part, the global atmospheric light estimation subnetwork, and the medium transmission estimation subnetwork. The haze-free image can be restored from the global atmospheric light and the medium transmission that achieved from the network. However, the method tends to amplify existing image artifacts and noise for some hazy scenes. Li *et al.* [35] used a residual-based deep CNN to dehazing. The network model has divided into two phases: the first stage, a haze image is an input to estimate the transmission map by the network; the second stage, the ratio of the foggy image and transmission map is used as input, and the residual network is used to train the atmospheric light and to dehazing. In this work, the NYU2 depth dataset [32] and RESIDE dataset [36] are used as training and test sets. Even though this model effectively performed dehazing processing for different scenes, especially in dark scenes, it still required a vast amount of images to training the model. In sum, although these methods enhance the haze-free results, the accuracy of the estimated transmission map and the dehazing

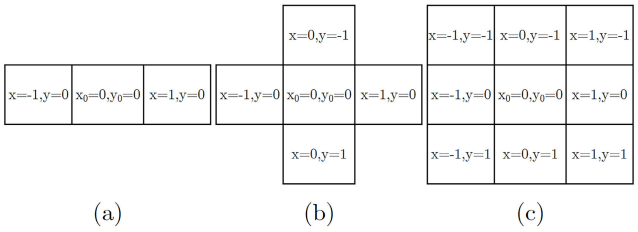

FIGURE 2. Three types of the neighborhood with $r = 1$: (a) 1-D neighborhood, (b) Von Neumann neighborhood, and (c) Moore neighborhood.

result need to be further improved. For instance, they are not very robust for the hazy image with sky region scene and heavily hazy scenes. Besides, they are also required a vast amount of images to train the model.

C. CELLULAR AUTOMATA

Cellular automaton (abbreviated CA, or Cellular Automata for plural) is firstly proposed by [37], [38] to describe the evolution process of dynamic complex system. CA is a dynamical model that time and space are discrete. It consists of a regular grid of cells in any finite number of dimensions. The set of cells (or called its neighborhood) is defined relative to the central cell. The next state of any cell is dependent on the rule in terms of the current state of the cell and its neighborhood. Cellular automata have various types of neighborhood (see Fig. 2).

The most commonly used of two-dimensional cellular automata are Moore and Von Neumann neighborhoods. The simplest class of one-dimensional cellular automata is proposed by Wolfram [39], called Elementary Cellular Automata (abbreviated ECA). ECA has two possible states and evolution rules, depending only on the nearest neighbor values. In this paper, we simulated on the neighborhood of Moore in which similar to the notion of 8-connected pixels of digital image structure [40] to extracting the feature of the image, and to enhancing the image contrast and color saturation of the haze-free image. More specifically, the neighborhood $N(x, y)$ is defined as follow:

$$N(x_0, y_0)^M = \{(x, y) : |x - x_0| \leq r, |y - y_0| \leq r\} \quad (11)$$

where M denoted Moore, N is a neighborhood, x and y are the image pixel or position of data in a matrix, x_0 and y_0 are the central point of the neighborhood, and r is radius or range of neighborhood.

$$N(x_0, y_0)^V = \{(x_0, y_{-1}), (x_{-1}, y_0), (x_0, y_0), (x_1, y_0), (x_0, y_1)\} \quad (12)$$

where V denoted Von Neumann, x and y are the coordinates of this neighborhood in the range $r = 1$.

$$N(x_0, y_0)^E = \{(x_{-1}, y_0), (x_0, y_0), (x_1, y_0)\} \quad (13)$$

where E denoted ECA or 1-D neighborhood in range $r = 1$.

CA have successfully been used in image processing, such as edge detection [41]–[44], noise filtering [40], [45]–[47], saliency detection [48]–[50], image segmentation [51]–[53], and 3D image reconstruction [54]. For example, Wongthanavasu and Sadananda [41] proposed an edge detection method based on a cellular automata model. In this work, a uniform cellular automaton rule using a von Neumann neighborhood has been used for carrying out the edge detection on binary and gray-scaled images. Jana *et al.* [55] applied the cellular automata in a noise filtering technique. The difference values of Moore neighbors from a center pixel, and all pixels value of Moore neighbor, including center pixel, are calculated. Then the values are sorted in ascending order to eliminated a minimum and maximum values and then updated the center pixel value using CA rule. Sahin *et al.* [40] and Qadir and Shoosha [46] proposed an image denoising algorithms to restore digital images corrupted by impulse noise. Both methods are based on two-dimensional cellular automata. Reference [40] used the cellular automata with the help of fuzzy logic theory, while [46] used the hybrid rules under null and periodic boundary conditions. Qin *et al.* [49] proposed an unsupervised Hierarchical Cellular Automata (HCA) to detect salient objects in the image. The HCA consists of two main components: Single-layer Cellular Automata (SCA) and Cuboid Cellular Automata (CCA). Single-layer Cellular Automata exploited the relevance of similar regions through interactions with neighbors. Low-level image features and high-level semantic information extracted from deep neural networks are used to measure the correlation between different image patches. The saliency maps will be iteratively updated according to well-defined update rules. The CCA integrates multiple saliency maps generated by SCA at different scales in a Bayesian framework to increase the performance of the model. Li *et al.* [53] proposed image segmentation method based on fuzzy clustering with cellular automata (CA) and features weighting. The method combined image color spatial feature weighting and the CA's self-iteration to speeds up the convergence of image segmentation. Sompong and Wongthanavasu [51] proposed a Gray-level co-occurrence matrix based cellular automata (GLCM-CA) framework and Improved Tumor-Cut (ITC) algorithm to cope with ambiguous tumor boundaries on brain tumor segmentation task. The GLCM-CA transformed an original magnetic resonance (MR) image to the target featured image, while the ITC used a patch weighted distance to enhances the robustness of seed growing. Olague *et al.* [54] proposed the infection algorithm based on an artificial epidemic process inspired by CA for 3D scene reconstruction. In this work, they present the Epidemic cellular automata that aim to match the contents of two images to obtain 3D information that allows the generation of simulated projections from a different viewpoint (also known as view synthesis).

For CA-based learning approaches, Wali and Saeed [56] proposed an ensemble learning architecture called Cellular Automata Learning and Prediction (CALP) model for the classification of handwritten patterns. The model allows the handwritten patterns to evolve or grow using various parameters that control by the cellular automata rules. Then these

different evolved patterns are used to train the classifier. Besides, the most related work was proposed by Nichele and Molund [57]. They proposed a deep learning framework with cellular automaton-based for reservoir computing. In this work, the cellular automata are used as a reservoir of data and tested on the 5-bits memory task (aka. well-known benchmark of the reservoir computing). The main objective of the model is to mapping the input binary pattern to the binary output correctly. The elementary cellular automata (or 1-D neighborhood) are used as the medium in the two reservoirs. In the encoding stage, the input is randomly mapped to initial data in the first row of the reservoir, then evolved by the cellular automata rule for the next row. In order to compute the output, they used linear regression model to interpret the readout value then fed it to the input of the next layer. The results show that the single CA reservoir system yields similar results to state-of-the-art, but the two-layered CA reservoirs show a noticeable improvement compared to a single CA reservoir.

IV. THE PROPOSED METHOD

In this section, we elaborate on the proposed method details: basics of DeepCA learner, CA's rule types used in this work, and DeepCA architecture. We then present the training of DeepCA and the building of the best parameter banks.

A. BASICS OF DEEP CELLULAR AUTOMATA LEARNER

Definition 1 (DeepCA Evolution Rule): Cellular automata work with the rule in general, the evolution rule is a necessary function that evolves the current state to the next state.

$$S_{ij}^{t+1} = f(S_{ij}^{t}, N_{ij}) \tag{14}$$

where S_{ij}^{t+1} is a next state of the i^{th} cell at j^{th} layer, f is the transition function, S_{ij}^{t} is a current state for the i^{th} cell at j^{th} layer, N_{ij} is neighborhood configuration of the i^{th} cell at j^{th} layer.

For single image haze removal task, we propose a rule suitable for the multi-layer cellular automata aiming to evolve any pixel to the haze pixel based on Eq. (14) as follow:

$$S_{ij}^{t+1} = f(S_{ij}^{t}, N_{ij}) \begin{cases} haze, & \text{if } pv_{ij}^{t} > cv_{threshold} \\ non-haze, & \text{otherwise} \end{cases} \tag{15}$$

where pv_{ij}^{t} is the mean of the pixel value in the neighborhood of the i^{th} cell at j^{th} layer at current state, and $cv_{threshold}$ is an confident value estimated from a group of the dark pixel in the dark channel, in which a current pixel is decided to a haze pixel.

For example, in range $r = 1$ of Moore neighborhood, the value of S_{ij}^{t+1} in any i^{th} cell at j^{th} layer obtained by Eq.(16) as follow:

$$\begin{aligned} S_{ij}^{t+1} = \frac{1}{9}(&a_0 S_{(x,y)}^{t} + a_1 S_{(x+1,y)}^{t} + a_2 S_{(x+1,y+1)}^{t} \\ &+ a_3 S_{(x,y+1)}^{t} + a_4 S_{(x-1,y+1)}^{t} \\ &+ a_5 S_{(x-1,y)}^{t} + a_6 S_{(x-1,y-1)}^{t} \\ &+ a_7 S_{(x,y-1)}^{t} + a_8 S_{(x+1,y-1)}^{t}) \end{aligned} \tag{16}$$

where $a_0, a_1, \ldots, a_8$ are the value of an image pixels in each neighbor of i^{th} cell.

Definition 2 (DeepCA Feature Matrices): DeepCA generated the multi-layer of data features using convolution function with the rule vector in each layer, the depth of these features enables to increase the classification accuracy. DeepCA feature matrices are shown in Eq.(17) as follow:

$$\begin{aligned} F_j &= f_{conv}(I_j, R_j, s), \\ &= \{f_{r_1}, f_{r_2}, \ldots, f_{r_n}\} \end{aligned} \tag{17}$$

where F_j denotes the feature matrices of j^{th} layer, $f_{conv}(I_j, R_j, s)$ represents the convolution function of input I in any dimension, and rule R at j^{th} layer with a stride number s, R_j denotes the rules vector at j^{th} layer represented as $<r_1, r_2, \ldots, r_n>$, and f_{ri} denotes the feature matrix obtained by the i^{th} rule (r_i), for $i = 1$ to n.

Definition 3 (Score Matrix (SM)): In the training process of DeepCA that aims to build the reference model, we built the memory section of DeepCA called *Score Matrix* (*SM*). It can be built corresponding to an original size of the feature matrices or modified the size as follow:

$$SM = f_{pool}(F_j, N_{ij}, s) \tag{18}$$

where SM denotes the score matrix of all feature layers, f_{pool} denotes the scoring function, e.g., $maxPool(y = max(x_{patch(i,j)}))$, $softMax(\sigma(z_j) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}})$ and $maxOut(y = max(x_{patch(i,j,k)}))$. F_j denotes the feature matrices of j^{th} layer, N^{ij} denotes the neighborhood configuration of the i^{th} cell at j^{th} layer, and s denotes a stride number.

Definition 4 (DeepCA Decision Rule): The structure of the score matrix of DeepCA is capable of decision or classification tasks. The decision rule for deciding a class of any input data is defined as the minimum error between the score matrix of an input image (SM_{r_i}) and the desired data class I_{class} as follow:

$$SM_{model} = min(f_{err}(SM_{r_i}, I_{class})) \tag{19}$$

where SM_{model} represents the score matrix of the model, and f_{err} represents an error estimation function of the score matrix, i.e., mean squared error (MSE).

Definition 5 (Rule Vector): The rule vector of DeepCA is the most important for the classification task because it is a major key to all DeepCA layers in terms of the score matrices values tuning. In the general type of cellular automata rules [58], the rule members are all rules in the Moore neighborhood space. For instance, the Moore neighborhood with two states possible given as follow:

$$R_j = < r_1, r_2, \ldots, r_n > \tag{20}$$

where R_j represents the rule vector at j^{th} layer, $r_1, r_2, \ldots, r_n$ denote the 1^{st} to n^{th} of rule members in the rule vector, r_i denote any rule number, i.e., rule-0 to rule-2^{2^9} (or $1.34e^{+154}$) for general rule type of Moore neighborhood [58].

Due to the vast rules space of the general rule type of Moore, we have to reduce the rules space to make it possible

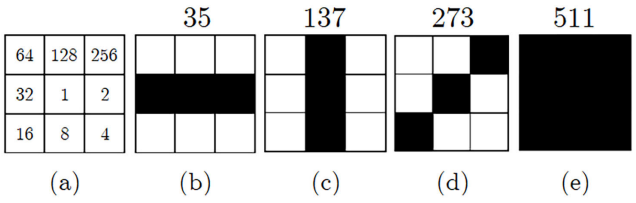

FIGURE 3. Totalistic rules: (a) The numbering of rule conventions with respect to neighbors, and (b)-(e) Example of rule-35, rule-137, rule-273, and rule-511, respectively.

to determine as the rule vector. In this regard, the totalistic rule type proposed by [58]–[60] is chosen to reduce the rules space of general rule type from all of 2^{2^9} rules to totalistic rule type as only 2^9 (or 512) possible rules (see Fig. 3). Fig. 3 (a) shows the specific setting of the Moore neighborhood in which the neighbors are in order of 2^n according to [55], [59], [60].

Definition 6 (The Equivalence of the General and the Totalistic Rule Types): In general, the cellular automata rule space is dependent on the type of neighborhood and possible states. For Moore neighborhood (size 3×3) with 2 possible states (0 or 1), the rule numbers in general rule type can be started from rule-0 to rule-$2^{512} - 1$ (see Fig. 4) while the totalistic rule type will be started only from rule-0 (neighborhood code "000000000") to rule-$2^9 - 1$ (rule-511, neighborhood code "111111111") (see Fig. 5). Actually, all of the totalistic rules are subset of the general rules and they have a significantly relation in term of their rule space, e.g., rule-35 in general rule type, the result of the rule depends on three neighborhood code that consists of "000000000", "000000001", and "000000101", respectively, whereas the result of the totalistic rule type depends on only single neighborhood code "000100011". In this regard, the equivalent of these rule types can be formalized as follow:

$$r_{totalistic}(x) \equiv r_{general}(2^{(x)}) \tag{21}$$

For example, the rule-35 can be represented in totalistic rule type as $r_{totalistic}(35) =$ "000100011" which equivalent to the general rule $r_{general}(2^{35}) =$ "000100011". On the other hand, the use of general rule $r_{general}(2^0 + 2^1 + 2^5)$ means that there are neighborhood codes "000000000", "000000001", and "000000101". They are also equivalent to $r_{totalistic}(0)$, $r_{totalistic}(1)$, and $r_{totalistic}(5)$, respectively.

Definition 7 (The Rule-0): Eventhrough the meaning of rule-0 is a rule that evolves any state to itself (or no operated), It still has a differences in the case of general rule type and totalistic rule type. In this work, we defined $r_{totalistic}(0) =$ "000000000", and $r_{general}(0) = null$.

Definition 8 (DeepCA Architecture): The major architecture of DeepCA is specially defined by multi-layer cellular automata, which can be formalized as follow:

$$F(x) = f_L(f_{L-1}(f_{L-2}(\ldots(f_1(x))))) \tag{22}$$

where $F(x)$ represents DeepCA's architecture function with input x, f_L represents functional layer of L^{th} layer that defined

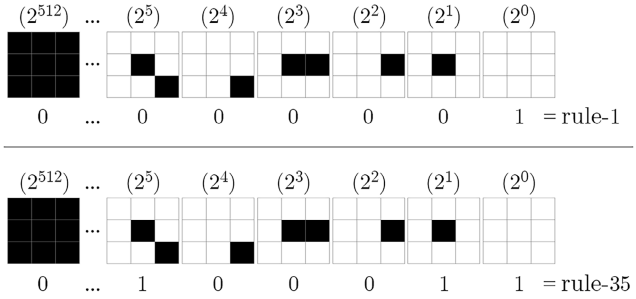

FIGURE 4. Example of general rules: (top) rule-1, and (bottom) rule-35, respectively.

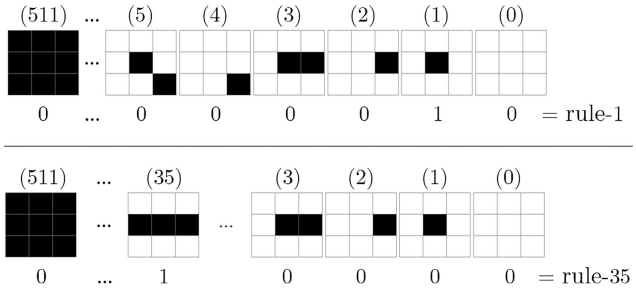

FIGURE 5. Example of totalistic rules: (top) rule-1, and (bottom) rule-35, respectively.

as input layer (f_{in}), convolution layer (f_{conv}), pooling layer (f_{pool}), and output layer (f_{out}).

B. PROPOSED FRAMEWORK

The main diagram of DeepCA architecture is illustrated in Fig. 6. In this regard, we consider DeepCA as multiple layers $Layer-1$ to $Layer-n$ and defined the Definition. 8 as $F(x) = f_{out}(f_{pool}(f_{pool}(f_{pool}(f_{conv}(f_{pool}(f_{conv}(f_{pool}(f_{conv}(f_{in}(x)))))))))$. In this architecture, function $f_{in}(.)$ is the first functional layer that separated each channel of the RGB image then feed to the next layer. Second layer is obtained by convolution function (f_{conv}) of the input image and the rule vector. It is to determine the data features that correspond to the rule called feature matrices. We then build a score matrix from these feature matrices using Eq. (18). The $maxPool(.)$ operation function is applied to all $f_{pool}(.)$, then the score matrix is determined to form by the functional layer (f_{out}) properly.

C. DeepCA TRAINING

DeepCA training process for each layer is illustrated in Fig. 7 correspondings to Algorithm 1. Firstly, the Moore neighborhood of size 3×3 and the rules vector are initialized. Then, the input image is separated into each RGB channel, and all pixels are evolved to the next state according to the convolution function and the rules vector defined by Eq. (14), Eq. (17), and Eq. (20), respectively. The results are formed to data features called the feature matrices F_j, that a number of the feature matrix f_{r_i} in F_j depends on a number of rules in the rules vector R_j (see Eq. (17) and Eq. (20)). Secondly,

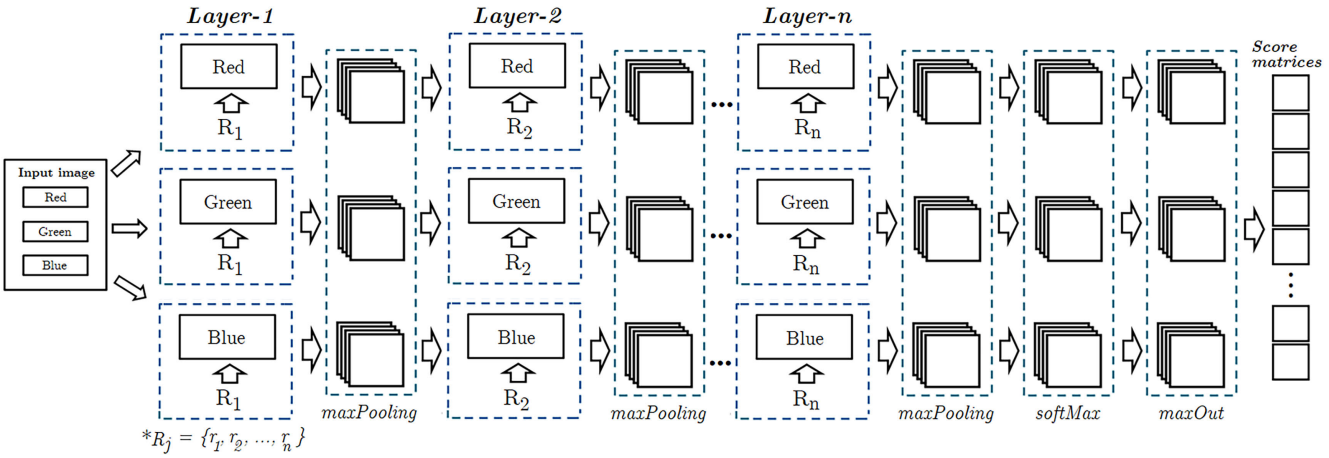

FIGURE 6. DeepCA architecture.

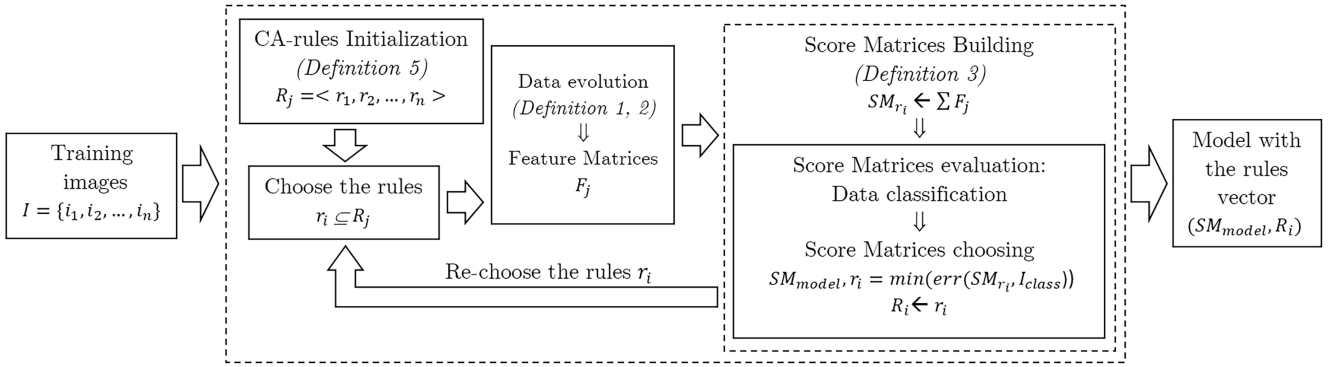

FIGURE 7. DeepCA training process.

these feature matrices are taken to the next layer or to build the score matrices SM, as shown in Eq. (18) and Fig. 6. In this regard, the *maxPool* function is applied on all layer-to-layer feature matrices, while the *softMax* and *maxOut* function will only be applied to the last layer. After *maxOut*, the score matrices can be extracted as an objective map. Thirdly, learning the mapping between input images and corresponding objective maps is learned by minimizing the loss function between the score matrices (SM_{r_i}) (or the predicted objective map) and the corresponding class of image (I_{class}) (or labeled data). We evaluated a minimum error of the model by the loss function (f_{err}) based on Eq. (19). The operation is repeated with all rules in the rules vector initialized by Eq. (20) for all training images until convergence. Finally, a mapping between the input image and the objective map is obtained by the score matrices.

D. DeepCA TRAINING FOR SINGLE IMAGE HAZE REMOVAL

DeepCA training process for single image haze removal is also illustrated in Fig. 7 correspondings to Algorithm 1. Firstly, the hazy image is separated into each RGB channel. Then the feature matrices are generated by evolving all pixels in each image channel to the next state and also evolving to the haze pixel by Eq. (14), and Eq. (15), respectively. In this layer, a number of feature matrices F_j in each image channel depends on a number of the rules r_i in the rules vector R_j. Secondly, the feature matrices are taken to the next layer or used to build the score matrices based on Eq. (18): function *maxPool*, *softMax* or *maxOut* will be applied depending on the layer in which it is located. Thirdly, learning the mapping between hazy images and corresponding transmission maps is learned by minimizing the loss function between the predicted transmission map (I_{pred}^{tran}) and the corresponding ground truth (I_{gt}^{tran}). We evaluated a minimum error of the model by the MSE loss function based on Eq. (19) as follow:

$$loss = \frac{1}{N}\sum_{i=1}^{N}(I_{pred}^{tran} - I_{gt}^{tran})^2 \tag{23}$$

where N is the number of each batch. Then, the rule number that provides the smallest error value will be registered in R_i. We repeated this process with all rules in the rules vector initialized by Eq. (20) for all training images until convergence. Finally, a mapping between the hazy image and the transmission map is obtained by the score matrices.

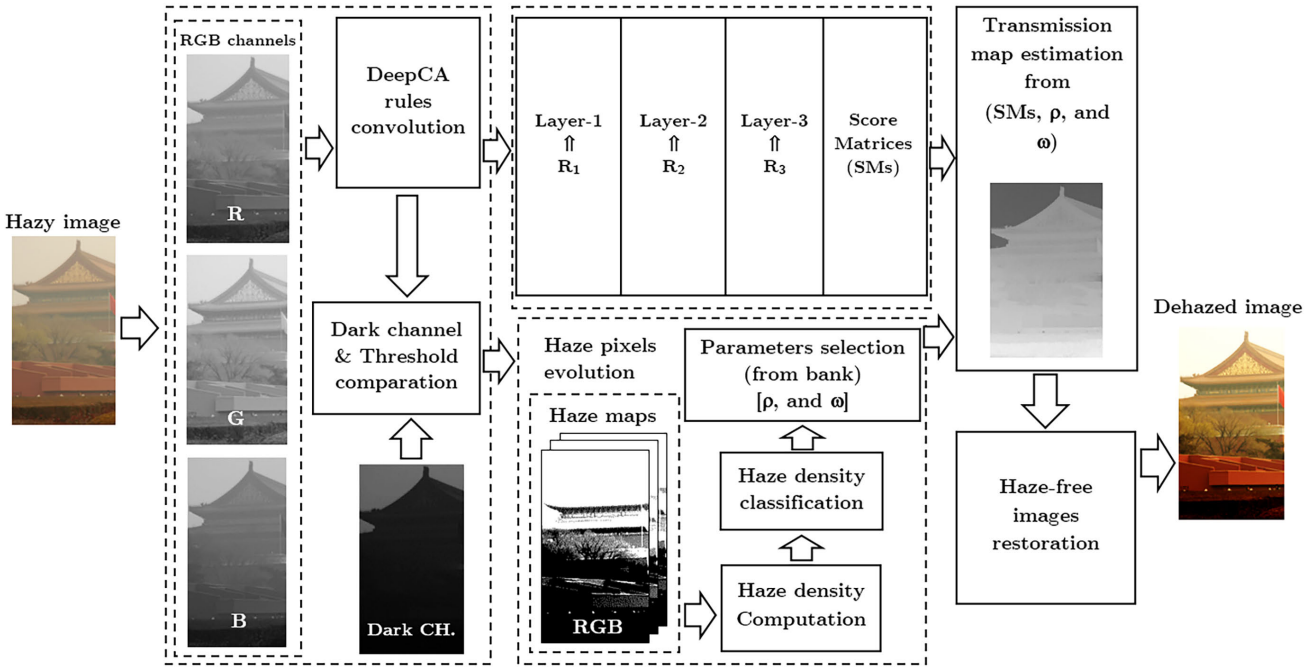

FIGURE 8. The architecture of DeepCA for single image haze removal.

E. DeepCA FOR SINGLE IMAGE HAZE REMOVAL

The diagram of the DeepCA for single image haze removal is illustrated in Fig. 8 corresponding to Algorithm 2. For more details, a hazy image is an input to DeepCA to generate the transmission map and to classify the haze density. It then uses the information of haze density to determine the best parameter of the haze preserve parameter (ω), and atmospheric light ratio (ρ) from the best parameters bank. These parameters provided by Algorithm 3 and Algorithm 4, they are the most important parameters to generate the best global atmospheric light. In this regard, the global atmospheric light value is achieved by Eq. (7) (p is set to 0.1), we then applied the haze preserve parameter (ω) from Eq.(9), and defined a new parameter (ρ) in Eq.(1) to adjust the ratio of global atmospheric light value resulting in Eq.(24) as follow:

$$I(x,y) = I'(x,y)t(x,y) + \frac{A(1-t(x,y))}{\rho} \tag{24}$$

Finally, the haze-free image $I'(x,y)$ is obtained by the global atmospheric light and the transmission map as follow:

$$I'(x,y) = \frac{I(x,y) - \frac{A(1-t(x,y))}{\rho}}{t(x,y)} \tag{25}$$

F. BEST PARAMETER BANK

In order to build the best parameter bank, we applied an Algorithm 3 to determine the best parameter for known ground-truth hazy image and applied an Algorithm 4 for unknown ground-truth hazy image. These algorithms are efficiently determining two significant parameters for dehazing. For the known ground-truth hazy image, the Algorithms 3 find the best values of both parameters in a range of $0 < \omega, \rho < 1$ using the MSE function to the evaluating between ground-truth image and the dehazed image provided by Algorithm 5. For the unknown ground-truth hazy image, the Algorithms 4 pre-defined parameters ρ_{pre} and ω_{pre} for temporarily determine the dark channel of a haze-free image provided by Algorithm 5 and the pre-defined parameters. The algorithm finds the best values of both parameters in a range of $0 < \omega, \rho < 1$ by evaluating the quality of the dark channel of temporary dehazed image and the dehazed image that also provided by Algorithm 5.

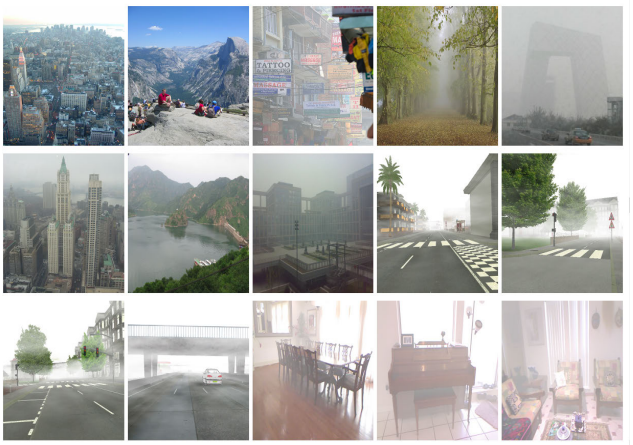

FIGURE 9. Example hazy images from R. Fattal [15], S. Wang *et al.* [61], FRIDA-1 and FRIDA-2 [62], [63], and RESIDE [36] datasets.

V. EXPERIMENTALS

In this section, we first describe the experimental settings and validate the proposed DeepCA on several datasets. Then, we

Algorithm 1 DeepCA Training Algorithm

Input: Images I and rule vector R_j.
Output: SM_{model}, and R_i: the class references model with rule vector.
Initialisation : 1
1: $N_{ij} \leftarrow 3 \times 3, R_j \leftarrow < r_1, r_2, \ldots, r_n >$ (Eq. (20))
LOOP Process : 2-13
2: **for each** Layer **do**
3: **while** !*convergence* **do**
4: **for all** image I **do**
5: **for each** $r_i \subseteq R_j$ **do**
6: Compute F_j by applying Eq. (14) and Eq. (17) on I
7: $SM_{r_i} \leftarrow f_{pool}(F_j, N_{ij}, s)$ (Eq. (18))
8: **end for**
9: $SM_{model} \leftarrow min(f_{err}(SM_{r_i}, I_{class}))$ (Eq. (19))
10: $R_i \leftarrow r_i$
11: **end for**
12: **end while**
13: **end for**
14: **return** SM_{model}, and R_i

Algorithm 2 DeepCA for Single Image Haze Removal

Input: Hazy images I, ρ, ω.
Output: Dehazed image I'.
Initialisation : 1
1: $N_{ij} \leftarrow 3 \times 3, R_j \leftarrow < r_1, r_2, \ldots, r_8 >$ (Eq. (20))
Process : 2-11
2: **for each** image I **do**
3: Compute F_j by applying Eq. (14), Eq. (15), and Eq. (17) on I
4: Compute $hazeDensity = \frac{(F_j)}{I_{width} * I_{height}}$
5: $SM_{R_j} \leftarrow hazeDensity$
6: $REF_{model} \leftarrow Min(MSE(SM_{R_j}, SM_{model}))$
7: $\rho, \omega \leftarrow Bank(REF_{model})$
8: Compute A by applying Eq. (7)
9: Compute I^{tran} by applying ω in Eq. (9)
10: Restore image I' by applying ρ in Eq. (25)
11: **end for**
12: **return** Dehazed image I'

Algorithm 3 Best Parameters Finding for Known Ground-Truth Hazy Image

Input: Hazy images I and its ground-truth G.
Output: $Bank(\rho, \omega)$, the bank of atmospheric light ratio ρ, and haze preserve value ω.
Initialisation : 1
1: $\rho, \omega \leftarrow 0$
LOOP Process : 2-9
2: **for all** image I **do**
3: **while** $\rho \leqslant 1$ **do**
4: **while** $\omega \leqslant 1$ **do**
5: $I' \leftarrow SHR(I, \rho, \omega)$
6: $\rho, \omega \leftarrow Min(MSE(I', G))$
7: **end while**
8: **end while**
9: **end for**
10: **return** $Bank(\rho, \omega)$

Algorithm 4 Best Parameters Finding for Unknown Ground-Truth Hazy Image

Input: Hazy images I, and pre-defined ρ_{pre} and ω_{pre}.
Output: $Bank(\rho, \omega)$: the bank of best atmospheric light ratio ρ, and haze preserve value ω.
Initialisation : 1-2
1: $\rho_{pre} \leftarrow 0.95, \omega_{pre} \leftarrow 0.85$
2: $\rho, \omega \leftarrow 0$
LOOP Process : 3-15
3: **for all** image I **do**
4: $I^{dark}_{pre} \leftarrow SHR(I, \rho_{pre}, \omega_{pre})$
5: **while** $\rho \leqslant 1$ **do**
6: **while** $\omega \leqslant 1$ **do**
7: $I^{dark} \leftarrow SHR(I, \rho, \omega)$
8: **if** $\Sigma(I^{dark}_{pre}) > \Sigma(I^{dark})$ **then**
9: $\rho, \omega \leftarrow getParam(I^{dark}_{pre})$
10: **else**
11: $\rho, \omega \leftarrow getParam(I^{dark})$
12: **end if**
13: **end while**
14: **end while**
15: **end for**
16: **return** $Bank(\rho, \omega)$

compare the dehazing results and the medium transmission with several state-of-the-art methods on both natural and synthetic benchmark images. In this regard, we directly use the dehazing source codes and the published results of the state-of-the-art method for the fairness of comparison.

A. DATASETS

For empirical experiments, three major groups of hazy images are implemented. The first group consists of 4 classes of 1464 natural images regarding the haze level proposed by Wang *et al.* [61]. The second group consists of 420 synthetic of foggy images and their ground truth (FRIDA-1 and FRIDA-2 datasets) proposed by Tarel *et al.* [62], [63], and the third group consists of 1,000 hazy images of Synthetic Objective Testing Set (SOTS) from the RESIDE dataset proposed by Li *et al.* [36]. Moreover, we also implemented all of the most popular images used by Fattal [15], and several state-of-the-art methods. These images consist of benchmark images, high-resolution images, ground truth images, and known transmission images. Fig. 9 shows examples of hazy images from these datasets.

B. TRAINING DATA

It is owing to a general costly to collect a huge amount of labeled data for training deep models [25], [29], especially for

Algorithm 5 Single Image Haze Removal, $SHR(I, \rho, \omega)$

Input: Hazy images I, ρ, ω.
Output: Dehazed image I'.
Initialisation : 1
1: $N_{ij} \leftarrow 3 \times 3$
Process : 2-7
2: **for each** image I **do**
3: Compute I^{dark} by applying Eq. (5)
4: Compute A by applying ρ in Eq. (7)
5: Compute I^{tran} by applying ω in Eq. (9)
6: Restore image I' by applying Eq. (4)
7: **end for**
8: **return** Dehazed image I'

pairs of clear images and haze images on natural images. For the training of DeepCA, we have synthesized the training data based on the dichromatic model [2]. The synthetic haze-free images of FRIDA-1 (18 images) and FRIDA-2 (66 images) datasets are used as training data for DeepCA. However, it is not enough to efficiently train the DeepCA. In this regard, we have randomly sampled 100 patches of size 50×50 on each image to DeepCA training: each patch is determined only the medium transmission based on the dichromatic model, and the atmospheric light (A) is set to 1 as suggested by [29], [35] to reduce the instability of the synthesis. Therefore, there are 8,400 of medium transmission patches generated for DeepCA training.

C. PERFORMANCE EVALUATION

We compared the proposed method and the state-of-the-art algorithms using Peak Signal to Noise Ratio (PSNR) [64] for quantitative evaluation. To compute the PSNR, we first calculate the mean squared error (MSE) using the following equation:

$$MSE = \frac{1}{N} \sum_{c \in \{r,g,b\}} (I'(c) - G(c))^2 \tag{26}$$

where $I'(c)$ is the haze-free image resulting from any algorithms, $G(c)$ is the ground-truth image, c is an image pixel at r, g and b color channel, and N is a number of image pixels.

PSNR represents a measure of the peak error. It is derived from the mean square error (MSE) and indicates the ratio of the maximum pixel intensity to the power of the distortion.

$$PNSR = 10 \log_{10}(\frac{(2^b - 1)^2}{MSE}) \tag{27}$$

where b is the bit size of a pixel of the image.

We also used the blind assessment method based on the property of the human visual system [65] and structural similarity (SSIM) measurement system [64] to objectively evaluate of the proposed method compared to the state-of-the-art methods on each of the single benchmark image. The SSIM is defined as:

$$SSIM(I', G) = \frac{(2\mu_{I'}\mu_G + C_1)(2\sigma_{I'G} + C_2)}{(\mu_{I'}^2 + \mu_G^2 + C_1)(\sigma_{I'}^2 + \sigma_G^2 + C_2)} \tag{28}$$

where I' represents a haze-free image, and G is a dehazed image. $2\mu_{I'}$ is the average of I', μ_G is the average of μ_G. $2\sigma_{I'G}$ is the covariance with I' and G. $\sigma_{I'}^2$ is the variance of I', and σ_G^2 is the variance of G. We set the values C_1 and C_2 to 0.01 and 0.03 by default as suggested in [64].

For the blind assessment method, it consists of three indicators: e_r is a ratio of edges newly visible after dehazing with local contrast above 5% of the hazy image and restored image, $\bar{r}$ is the ratio of the quality of contrast after and before dehazing, and σ_s is the percentage of pixels which become saturated after dehazing.

D. TRAINING RESULTS ON HAZE REMOVAL PROBLEM

The problem statement in Section II mentions the dichromatic model and the importance of ω and ρ. These parameters do not only affect the transmission map but also significantly influence on the resulting of haze-free image restoration. Hence, we proposed the DeepCA learning to the mapping between the hazy images and their proper transmission map. Regarding the training datasets, Frida-1 and Frida-2 consist of the haze-free images and four types of haze that integrated in the image, i.e., homogeneous fog, heterogeneous fog, cloudy homogeneous fog, and cloudy heterogeneous fog. The haze-free images are used to estimating the transmission map. We then used these transmission maps to train the DeepCA model. Fig. 7 illustrated the training process of DeepCA corresponding to Algorithm 1 that described the training step of each layer on DeepCA architecture. Meanwhile, Fig. 10 (a)-(c) depict the values of training error and a number of the rules used on DeepCA layer-1, layer-2, and layer-3, respectively.

E. QUANTITATIVE RESULTS

To quantitatively verify the performance of the proposed DeepCA, we validated DeepCA on both natural hazy and synthetic hazy images by using evaluation metrics based on the difference between a pair of their hazy or haze-free images, and dehazing result. For natural hazy images, we used images from Fattal [15] and Wang *et al.* [61] to verify the model performance. The blind assessment method, e_r, $\bar{r}$, and σ_s [65] are applied to evaluate the original hazy image and dehazing result. The quantitative results are provided in Table. 1. As shown, the proposed DeepCA achieves better performances of e_r on most of the images in the first rank and the second rank meaning that DeepCA appears to be capable of recovering the new edge visible better than others. For $\bar{r}$, DeepCA also achieves the quality of contrast in the first and the second rank on most of the images. For σ_s, the value that closes to zero signifies the better performance. DeepCA is able to maintain the percentage of pixel saturation compared to other methods.

In the case of synthetic hazy images (FRIDA and SOTS datasets), we used the PSNR and the SSIM to evaluate the haze-free image (or ground-truth) and dehazing result. The average values of these evaluation metrics obtained by DeepCA and the state-of-the-art compared methods are listed

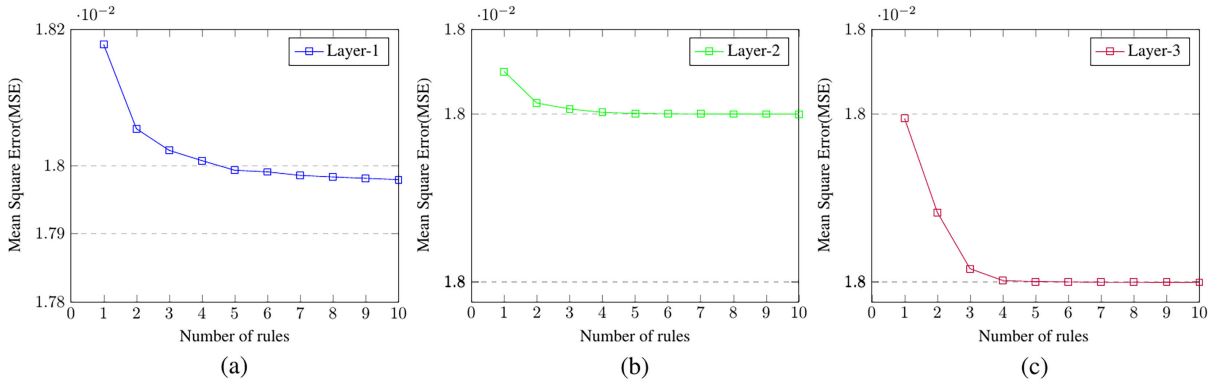

FIGURE 10. Training process in Layer-1, Layer-2, and Layer-3 with the different number of the rules: (a)-(c) MSE of Layer-1, Layer-2, and Layer-3, respectively.

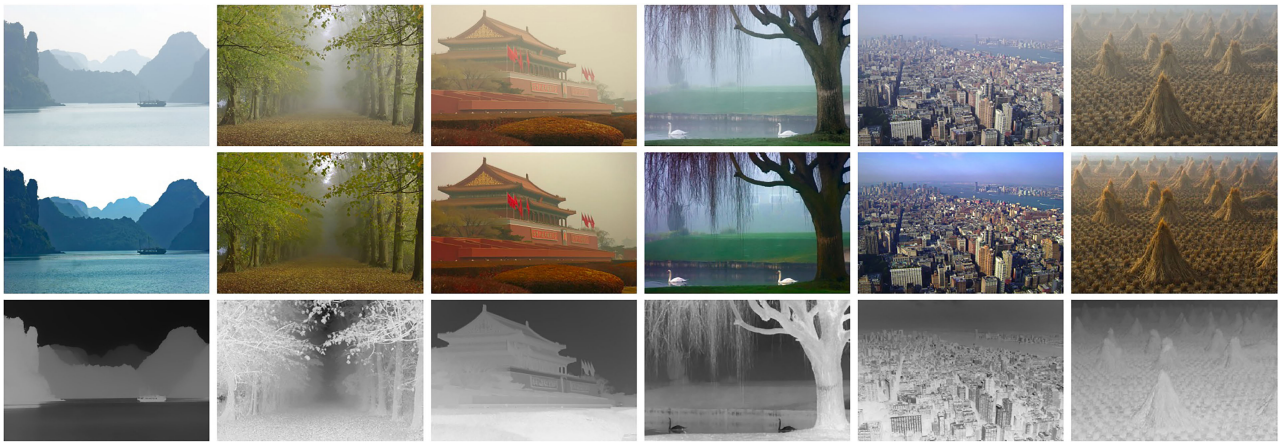

FIGURE 11. Dehazing results of the natural images and their transmission maps restored by DeepCA.

and illustrated in Table. 2, Table. 3, Fig. 12, and Fig. 13, respectively. For the Frida datasets, the proposed DeepCA performs competitively against state-of-the-art algorithms. The use of synthetic haze-free images and its transmission maps in the training process allows DeepCA to produce the best PSNR and SSIM, means that the proposed method is able to significantly maintain the image content and the similarity of the structures of the original image. For the SOTS dataset (indoor and outdoor images), we also compared the dehazing performance with other learning-based dehazing methods. Table 3. shows that the DeepCA performs relatively high against the learning-based dehazing methods in terms of the average PSNR and SSIM values. Although DeepCA has achieved the third and second rank for indoor images, the PSNR and SSIM generated by DeepCA are higher than the He *et al.* [13], Zhu *et al.* [19], Choi *et al.* [11], Bui and Kim [66], and Li *et al.* [36] methods. For outdoor images, Cai *et al.* [29], Zhu *et al.* [19], and Ren *et al.* [31] obtain greater PSNR advantages over all methods. However, the DeepCA also performs relatively high PSNR and SSIM compared to other methods.

F. QUALITATIVE RESULTS ON NATURAL IMAGES

To evaluate the performance of the proposed DeepCA and the state-of-the-art compared algorithms, we simulated these algorithms on the benchmark hazy images provided by Fattal [15], and Wang *et al.* [61]. Fig. 11 shows the hazy images, dehazing results, and the corresponding transmission maps generated by the DeepCA. As can be seen, the DeepCA is able to significantly recover haze-free pixels from the hazy images of various image scenes and preserve the subtle transitions in the hazy regions without introducing the halo artifacts.

The sky region in images is mentioned as the dehazing problem, especially in cloudy landscape scenes [11], [13], and [29], because of two reasons: haze and clouds are similar to the color in natural phenomenons, and the proportion of the sky or the clouds in the image can cause under-saturation or over-saturation of haze-free image scenes restoration. Fig. 14 shows the result of images with sky region Ny17 and Yos2 that focused on the red marked region. It can be seen that the DeepCA appropriately produced haze-free images of the landscape scenes without under-saturated or over-saturated compared to other methods. Meanwhile, the hazy from the results of Zhu *et al.* [19], Cai *et al.* [29], and Ren *et al.* [31] has not completely reduced. In addition, the dehazed images from Bui and Kim [66], and Li *et al.* [36] methods are tend to become over-saturation.

Fig. 15 shows the qualitative comparison of DeepCA and the state-of-the-art methods on the most popular benchmark

TABLE 1. Comparison rate e_r, $\bar{r}$, and σ_s of natural images with the state-of-the-art methods. Note Text color blue = 1^{st} rank, magenta = 2^{nd} rank, and green = 3^{rd} rank.

	Images	He et al. [13]	Zhu et al. [19]	Choi et al. [11]	Cai et al. [29]	Bui et al. [66]	DeepCA
e_r	Mahattan	0.222	0.235	0.136	0.045	0.140	0.256
	canon	1.234	0.928	1.362	1.527	4.767	1.970
	cones	0.207	0.109	0.190	0.082	0.296	0.286
	flags	0.347	0.221	0.224	0.246	0.189	0.391
	forest	0.357	0.155	0.329	0.160	0.445	0.501
	house	0.147	0.094	0.173	0.084	0.116	0.184
	mountain	0.210	0.069	0.164	0.111	0.173	0.256
	pumpkins	0.188	0.072	0.205	0.091	0.439	0.266
	swan	0.480	0.173	0.418	0.184	0.734	0.709
	tiananmen	0.363	0.140	0.457	0.224	0.604	0.537
	train	0.897	0.793	0.904	1.010	1.527	1.516
	tree	1.391	0.760	0.498	0.863	2.006	2.782
	yos1	0.246	0.072	0.088	0.029	0.145	0.249
	yos2	0.206	0.105	0.125	0.070	0.100	0.214
$\bar{r}$	Mahattan	1.324	1.346	1.468	1.180	0.788	1.480
	canon	1.690	1.588	1.879	1.701	3.343	1.941
	cones	1.314	1.189	1.676	1.341	1.560	1.474
	flags	1.816	1.617	1.691	1.620	2.806	2.164
	forest	1.262	1.092	1.408	1.193	1.223	1.421
	house	1.290	1.189	1.606	1.106	1.796	1.419
	mountain	1.208	1.164	1.224	1.222	1.273	1.312
	pumpkins	1.294	1.159	1.508	1.325	1.729	1.386
	swan	1.419	1.280	1.462	1.223	1.626	1.482
	tiananmen	1.307	1.178	1.327	1.223	1.553	1.624
	train	1.369	1.277	1.703	1.340	1.780	1.554
	tree	1.990	1.680	1.662	1.753	2.044	2.554
	yos1	1.267	1.143	1.550	1.144	0.895	1.628
	yos2	1.489	1.244	1.420	1.182	1.340	1.640
σ_s	Mahattan	0.001	0.001	0.003	0.000	0.001	0.001
	Canon	0.000	0.000	0.000	0.000	0.000	0.000
	Cones	0.000	0.000	0.001	0.000	0.000	0.000
	Flags	0.000	0.000	0.018	0.007	0.005	0.000
	Forest	0.001	0.000	0.009	0.000	0.001	0.001
	House	0.000	0.000	0.013	0.001	0.000	0.000
	Mountain	0.000	0.000	0.001	0.000	0.000	0.000
	Pumpkins	0.000	0.000	0.001	0.000	0.000	0.000
	Swan	0.000	0.000	0.074	0.004	0.000	0.011
	Tiananmen	0.000	0.000	0.038	0.007	0.000	0.000
	Train	0.001	0.000	0.027	0.024	0.002	0.010
	Tree	0.000	0.000	0.000	0.000	0.000	0.001
	Yos1	0.001	0.000	0.000	0.002	0.001	0.001
	Yos2	0.000	0.000	0.004	0.001	0.001	0.000

Note Text color blue = 1^{st} rank, magenta = 2^{nd} rank, and green = 3^{rd} rank.

TABLE 2. Comparison of PSNR and SSIM on FRIDA datasets obtained by the state-of-the-art methods and DeepCA.

Datasets	Metrics	He et al. [13]	Zhu et al. [19]	Choi et al. [11]	Cai et al. [29]	Bui et al. [66]	DeepCA
FRIDA-1	PSNR	13.458	13.531	13.195	12.889	11.998	13.663
	SSIM	0.568	0.602	0.580	0.530	0.487	0.639
FRIDA-2	PSNR	11.920	12.188	12.035	11.793	10.812	12.335
	SSIM	0.623	0.706	0.726	0.637	0.530	0.731

images. Fig. 15 (a) depicts the original hazy image, (b)-(f) illustrate the results of He *et al.* [13], Zhu *et al.* [19], Cai *et al.* [29], Bui and Kim [66], Li *et al.* [36], Ren *et al.* [31] and (g) shows the results of DeepCA, respectively. As can be seen, all dehazing algorithms can gain good results in general outdoor images except for images that come with the sky, the cloud or the large white areas that some dehazing methods cannot handle. For instance, the dehazed images produced by Cai *et al.* [29] and Ren *et al.* [31] are look lightly dehazed (e.g., Man, Yos2, and Guogong) while Bui and Kim [66] produced the images that look over-saturated and still have some color distortions, but DeepCA obtain more natural

TABLE 3. Comparison of PSNR and SSIM on SOTS datasets obtained by the state-of-the-art methods and DeepCA.

Datasets	Metrics	He et al. [13]	Zhu et al. [19]	Choi et al. [11]	Cai et al. [29]	Bui et al. [66]	Li. et al. [36]	Ren. et al. [31]	DeepCA
Indoor	PSNR	16.620	19.050	17.255	21.140	15.107	19.060	22.300	20.340
	SSIM	0.818	0.836	0.770	0.847	0.713	0.850	0.880	0.857
Outdoor	PSNR	19.130	22.006	18.917	22.460	12.779	20.290	21.550	20.169
	SSIM	0.815	0.881	0.808	0.851	0.603	0.877	0.844	0.862

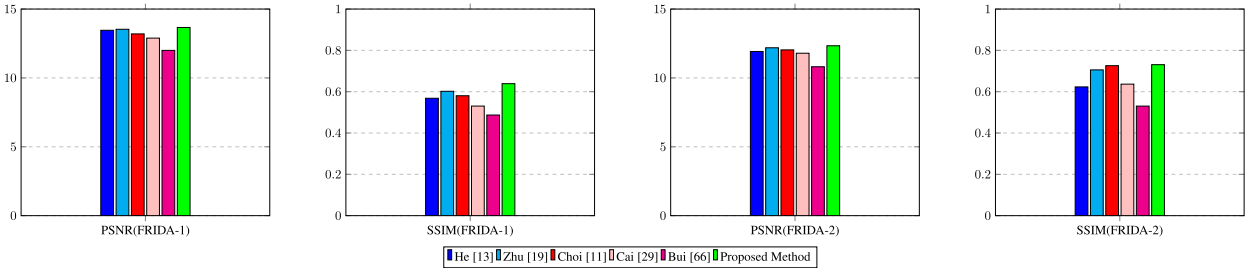

FIGURE 12. Comparison of PSNR and SSIM on FRIDA datasets obtained by state-of-the-art methods and DeepCA.

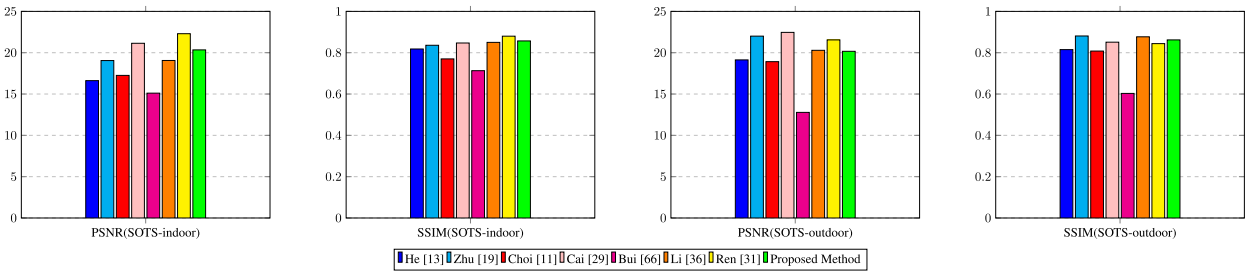

FIGURE 13. Comparison of PSNR and SSIM on SOTS datasets obtained by state-of-the-art methods and DeepCA.

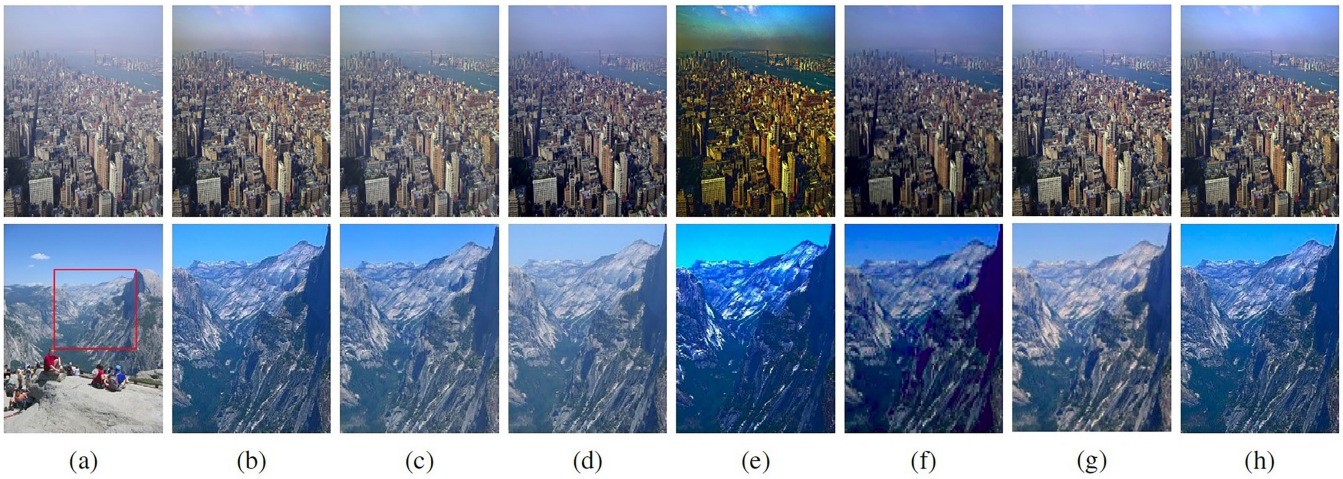

FIGURE 14. Comparison of dehazed images on landscape scene with the sky region of image Ny17, and image Yos2 in the red marked region: (a) Original hazy image, (b) Result of He *et al.*'s method [13], (c) Result of Zhu *et al.*'s method [19], (d) Result of Cai *et al.*'s method [29], (e) Result of Bui *et al.*'s method [66], (f) Result of Li *et al.*'s method [36], (g) Result of Ren *et al.*'s method [31] and (h) DeepCA.

results. For images House, He *et al.* [13] and Zhu *et al.*'s method [19] tent to produced the images that the hazy and the halo effect are not completely removed, while other methods and DeepCA do not suffer those problems.

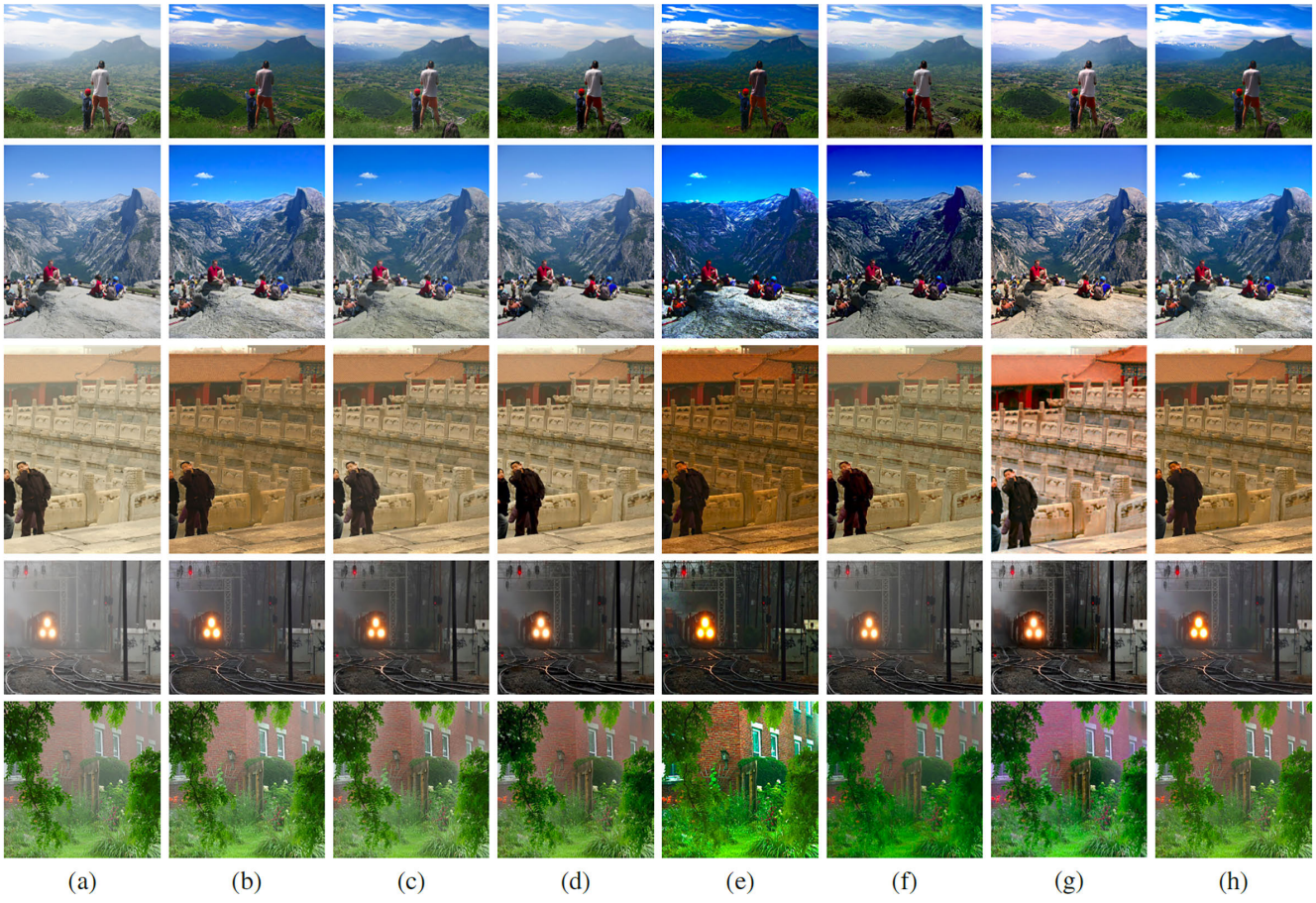

FIGURE 15. Qualitative comparison of dehazed images (Man, Yos2, Guogong, Train, and House) of the state-of-the-art methods and DeepCA: (a) Original hazy image, (b) Result of He *et al.*'s method [13], (c) Result of Zhu *et al.*'s method [19], (d) Result of Cai *et al.*'s method [29], (e) Result of Bui *et al.*'s method [66], (f) Result of Li *et al.*'s method [36], (g) Result of Ren *et al.*'s method [31] and (h) DeepCA.

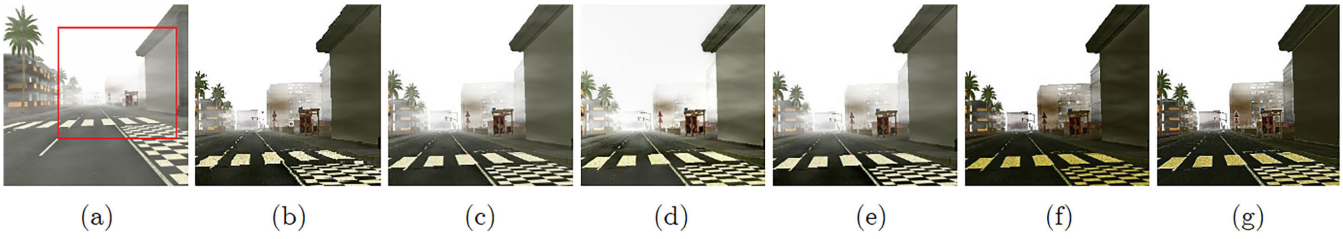

FIGURE 16. Comparison of object restoration on a synthetic hazy image from FRIDA dataset in the red marked region: (a) Original hazy image, (b) Result of He *et al.*'s method [13], (c) Result of Zhu *et al.*'s method [19], (d) Result of Choi *et al.*'s method [11], (e) Result of Cai *et al.*'s method [29], (f) Result of Bui *et al.*'s method [66], and (g) DeepCA.

G. QUALITATIVE RESULTS ON SYNTHETIC IMAGES

To evaluate the qualitative performance of the proposed DeepCA and the compared methods on synthetic images, we simulated the algorithms on FRIDA and SOTS datasets. For the Frida dataset, Fig. 16 shows the performance of DeepCA capable of obviously recovering the building object visible in the red marked region better than other methods, while the hazy from the results of Zhu *et al.* [22], Choi *et al.* [11], and Cai *et al.* [29] has not entirely removed. Fig. 17 shows the qualitative results in comparison of DeepCA and the state-of-the-art methods. Fig. 17 (a) depicts the original hazy image, (b)-(f) illustrate the results of He *et al.* [13], Zhu *et al.* [19], Choi *et al.* [11], Cai *et al.* [29], Bui and Kim [66], and (g) shows the results of DeepCA, respectively. It can be seen that the proposed DeepCA and Bui *et al.*'s method obviously produced the greatest reduction of haze density and yielded better results meaning that main objects can be restored from the dense-haze scenes, while other methods can only be reduced lightly haze. However, Bui and Kim [66] and DeepCA tend to produce results that contain distorted

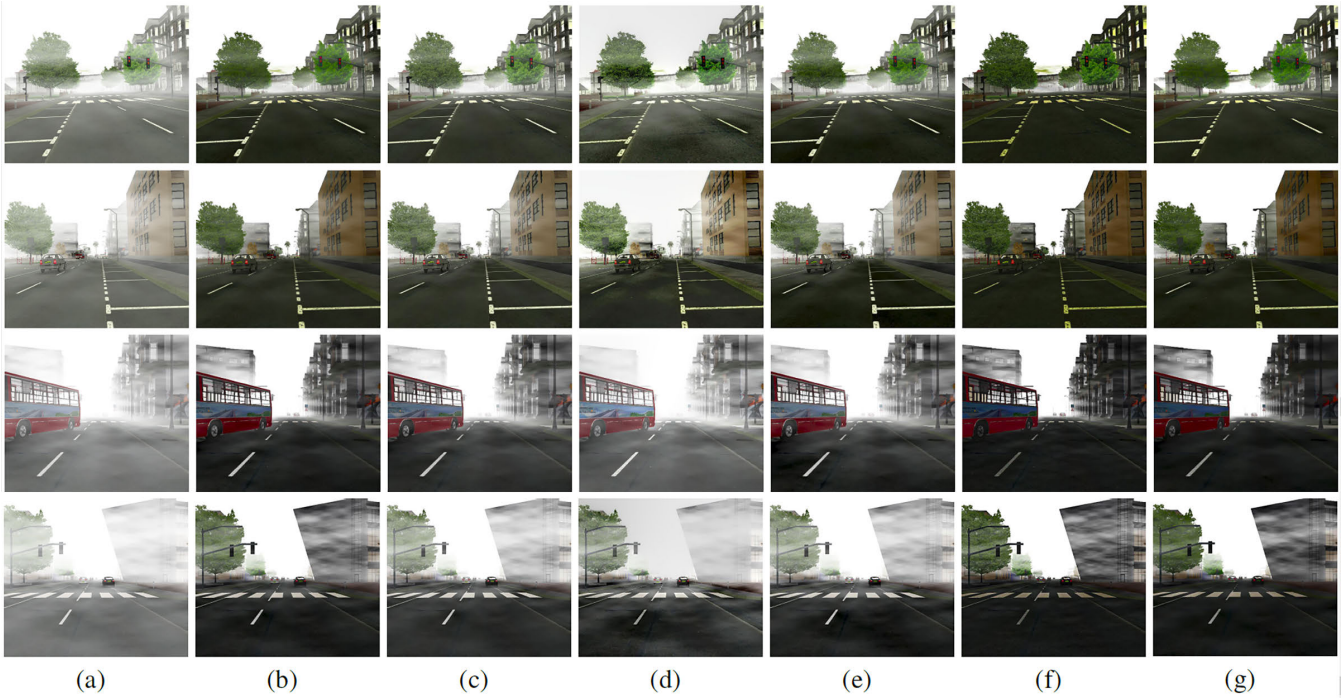

FIGURE 17. Qualitative comparison of dehazed images (on FRIDA datasets) of the state-of-the-art methods and DeepCA: (a) Original hazy image, (b) Result of He *et al.*'s method [13], (c) Result of Zhu *et al.*'s method [19], (d) Result of Choi *et al.*'s method [11], (e) Result of Cai *et al.*'s method [29], (f) Result of Bui *et al.*'s method [66], and (g) DeepCA.

FIGURE 18. Qualitative comparison of dehazed images (on synthetic indoor images from SOTS dataset) of the state-of-the-art methods and DeepCA: (a) Original hazy image, (b) Result of He *et al.*'s method [13], (c) Result of Zhu *et al.*'s method [19], (d) Result of Cai *et al.*'s method [29], (e) Result of Bui *et al.*'s method [66], (f) Result of Li *et al.*'s method [36], (g) Result of Ren *et al.*'s method [31] and (h) DeepCA.

colors for dense-haze scenes. For the SOTS indoor dataset, the comparison results are shown in Fig. 18. It can be seen that the results generated by He *et al.* [13], and Bui and Kim [66] suffer from color distortion where the results are usually darker than other methods. The methods of Zhu *et al.* [22], Cai *et al.* [29], and Li *et al.* [36] produce results that there remains some haze in the dehazing images. Meanwhile, Ren *et al.* [31] and DeepCA obtain the proper reduction of haze density and yielded better results. In the case of the outdoor dataset, Fig. 19 shows that most of the haze is removed by He *et al.* [13], Zhu *et al.* [19], Cai *et al.* [29], Li *et al.* [36], Ren *et al.* [31], and DeepCA. It can be seen that the results look very natural except for Bui and Kim [66]'s method that often generated over-saturated images and contains distorted colors. Comparing to the results of the state-of-the-art algorithms, DeepCA tends to significantly reduce the dense haze and achieves better visibility enhancement images on challenging images.

H. RUNNING TIME

Table 4 reports the average running time of the algorithms based on their codes published on the internet. We use the 100 images (640×480 pixels) in the FRIDA dataset for evaluation. All methods are implemented in MATLAB r2018a based on their source code available on the internet. We evaluate these methods on the same machine without GPU

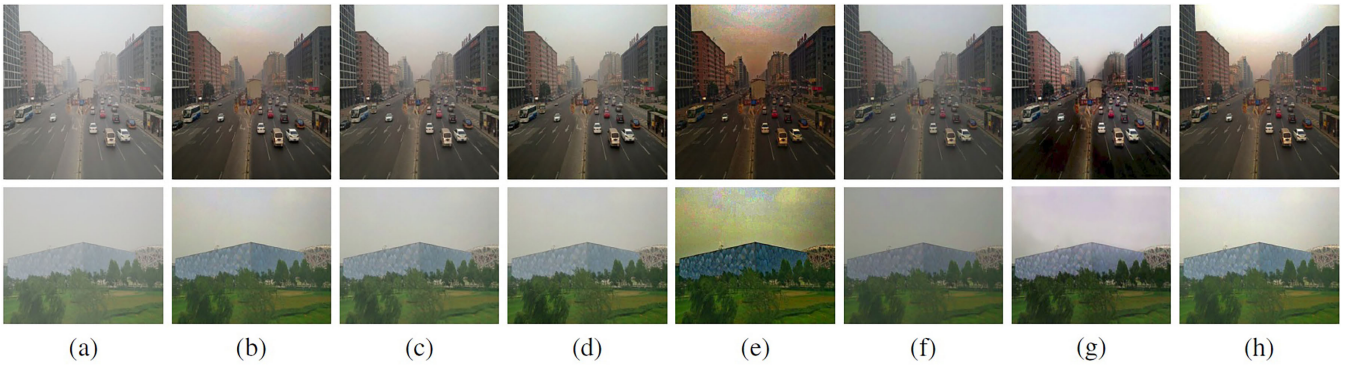

FIGURE 19. Qualitative comparison of dehazed images (on synthetic outdoor images from SOTS dataset) of the state-of-the-art methods and DeepCA: (a) Original hazy image, (b) Result of He *et al.*'s method [13], (c) Result of Zhu *et al.*'s method [19], (d) Result of Cai *et al.*'s method [29], (e) Result of Bui *et al.*'s method [66], (f) Result of Li *et al.*'s method [36], (g) Result of Ren *et al.*'s method [31] and (h) DeepCA.

TABLE 4. Comparison of the average running time (in seconds) on test image size of 640 × 480 pixels.

	He et al. [13]	Zhu et al. [19]	Choi et al. [11]	Cai et al. [29]	Bui et al. [66]	DeepCA
Time	25.36	0.87	28.52	2.73	0.19	6.79

acceleration (Intel CPU 3.60 GHz and 8 GB memory). The proposed algorithm is more efficient than the state-of-the-art methods (e.g., He *et al.* [13] and Choi *et al.* [11]) in terms of run time. However, the proposed method is slower than the other learning-based dehazing methods.

VI. CONCLUSION

We have presented a novel DeepCA that combines ideas of deep learning and cellular automata approach for improving single image haze removal. DeepCA learning is divided into two major parts: the first part is cellular automata deep feature extraction, we used multi-layers cellular automata with rules vector to extract the light source feature of hazy images, then this feature is formalized as score matrices. It was trained by the cellular automata rules to determine the proper transmission map and to estimate the haze pixels in the hazy image. The second part is a decision stage: we used the score matrices in mapping between the proper transmission map and hazy image and in deciding the haze density class of the hazy image. Then, the haze preserved parameter (ω) and the ratio of global atmospheric light value (ρ) are determined from the haze density class. This provides the parameters to enhance the transmission map suitable for restoring the best haze-free images. For performance evaluation, we also used the most popular natural benchmark images, high-resolution images, ground truth images, and synthetic images in the experiments to compare with the state-of-the-art algorithms. The simulation shows that the proposed DeepCA provides promising performance, improving image intensity, reducing the halo artifact, and obviously producing the most significant reduction of haze density when compared with the state-of-the-art algorithms.

Even though the proposed method significantly reduced the haze density in the image, especially in the dense-haze scenes, the method tends to amplify existing image artifacts for some image scenes, and the background details or color of some objects can be corrupted by noise on heavily hazy images. For future work, we intend to suppress these drawbacks in a further dehazing model. Additionally, we will improve the proposed model to directly estimation of the medium transmission (without any parameters) and also increase the speed of the dehazing process.

REFERENCES

[1] S. Natelson, *Aerosols and Atmospheric Chemistry*, G. M. Hidy, Ed. New York, NY, USA: Academic, 1972, p. 348.
[2] S. G. Narasimhan and S. K. Nayar, "Vision and the atmosphere," *Int. J. Comput. Vis.*, vol. 48, no. 3, pp. 233–254, 2002.
[3] Y. Y. Schechner, S. G. Narasimhan, and S. K. Nayar, "Polarization-based vision through haze," *Appl. Opt.*, vol. 42, no. 3, pp. 511–525, Jan. 2003. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-42-3-511
[4] Y. Y. Schechner, S. G. Narasimhan, and S. K. Nayar, "Instant dehazing of images using polarization," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2001, pp. 325–332.
[5] S. G. Narasimhan and S. K. Nayar, "Contrast restoration of weather degraded images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 6, pp. 713–724, Jun. 2003.
[6] S. K. Nayar and S. G. Narasimhan, "Vision in bad weather," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 820–827.
[7] J. A. Stark, "Adaptive image contrast enhancement using generalizations of histogram equalization," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 889–896, May 2000.
[8] K. Santhini, S. Priya, and T. Kumar, "Visibility enhancement of dimmed and hazy images using adaptive gamma correction along with automatic noise removal," *Int. J. Res. Inf. Technol*, vol. 3, no. 7, pp. 161–174, 2015.
[9] Z. Mi, Y. Zheng, H. Zhou, and M. Wang, "Single image dehazing via multi-scale gradient domain contrast enhancement," *IET Image Process.*, vol. 10, no. 3, pp. 206–214, Mar. 2016.
[10] C. O. Ancuti and C. Ancuti, "Single image dehazing by multi-scale fusion," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 3271–3282, Aug. 2013.
[11] L. Kwon Choi, J. You, and A. C. Bovik, "Referenceless prediction of perceptual fog density and perceptual image defogging," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3888–3901, Nov. 2015.
[12] E. H. Land, "The retinex theory of color vision," *Sci. Amer.*, vol. 237, no. 6, pp. 108–129, 1977.
[13] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.
[14] R. T. Tan, "Visibility in bad weather from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[15] R. Fattal, "Dehazing using color-lines," *ACM Trans. Graph.*, vol. 34, no. 1, pp. 1–14, Dec. 2014.
[16] S. Lee, S. Yun, J.-H. Nam, C. S. Won, and S.-W. Jung, "A review on dark channel prior based image dehazing algorithms," *EURASIP J. Image Video Process.*, vol. 2016, no. 1, p. 4, Dec. 2016.
[17] C. Xiao and J. Gan, "Fast image dehazing using guided joint bilateral filter," *Vis. Comput.*, vol. 28, nos. 6–8, pp. 713–721, Jun. 2012.
[18] J. Yu, C. Xiao, and D. Li, "Physics-based fast single image fog removal," in *Proc. IEEE 10th Int. Conf. Signal Process.*, Oct. 2010, pp. 1048–1052.
[19] Q. Zhu, J. Mai, and L. Shao, "A fast single image haze removal algorithm using color attenuation prior," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3522–3533, Nov. 2015.
[20] H. Xu, J. Guo, Q. Liu, and L. Ye, "Fast image dehazing using improved dark channel prior," in *Proc. IEEE Int. Conf. Inf. Sci. Technol.*, Mar. 2012, pp. 663–667.
[21] Z. Li and J. Zheng, "Single image de-hazing using globally guided image filtering," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 442–450, Jan. 2018.
[22] Y. Zhu, G. Tang, X. Zhang, J. Jiang, and Q. Tian, "Haze removal method for natural restoration of images with sky," *Neurocomputing*, vol. 275, pp. 499–510, Jan. 2018.
[23] R. Fattal, "Single image dehazing," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 1–9, 2008.
[24] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[27] A. Alani, "Arabic handwritten digit recognition based on restricted Boltzmann machine and convolutional neural networks," *Information*, vol. 8, no. 4, p. 142, Nov. 2017.
[28] S. Chen, H. Liu, X. Zeng, S. Qian, J. Yu, and W. Guo, "Image classification based on convolutional denoising sparse autoencoder," *Math. Problems Eng.*, vol. 2017, pp. 1–16, Nov. 2017.
[29] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "DehazeNet: An end-to-end system for single image haze removal," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5187–5198, Nov. 2016.
[30] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "AOD-net: All-in-one dehazing network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4770–4778.
[31] W. Ren, L. Ma, J. Zhang, J. Pan, X. Cao, W. Liu, and M.-H. Yang, "Gated fusion network for single image dehazing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3253–3261.
[32] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 746–760.
[33] W. Ren, S. Liu, H. Zhang, J. Pan, X. Cao, and M.-H. Yang, "Single image dehazing via multi-scale convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 154–169.
[34] C. Li, J. Guo, F. Porikli, H. Fu, and Y. Pang, "A cascaded convolutional neural network for single image dehazing," *IEEE Access*, vol. 6, pp. 24877–24887, 2018.
[35] J. Li, G. Li, and H. Fan, "Image dehazing using residual-based deep CNN," *IEEE Access*, vol. 6, pp. 26831–26842, 2018.
[36] B. Li, W. Ren, D. Fu, D. Tao, D. Feng, W. Zeng, and Z. Wang, "Benchmarking single-image dehazing and beyond," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 492–505, Jan. 2019.
[37] S. M. Ulam, "Some ideas and prospects in biomathematics," *Annu. Rev. Biophys. Bioeng.*, vol. 1, no. 1, pp. 277–292, Jun. 1972.
[38] J. Neumann, *Theory of Self-Reproducing Automata*, vol. 1102024. Champaign, IL, USA: Univ. of Illinois Press, 1966.
[39] S. Wolfram, "Computation theory of cellular automata," *Commun. Math. Phys.*, vol. 96, no. 1, pp. 15–57, 1984.
[40] U. Sahin, S. Uguz, and F. Sahin, "Salt and pepper noise filtering with fuzzy-cellular automata," *Comput. Electr. Eng.*, vol. 40, no. 1, pp. 59–69, Jan. 2014.
[41] S. Wongthanavasu and R. Sadananda, "A CA-based edge operator and its performance evaluation," *J. Vis. Commun. Image Represent.*, vol. 14, no. 2, pp. 83–96, Jun. 2003.
[42] T. Kumar and G. Sahoo, "A novel method of edge detection using cellular automata," *Int. J. Comput. Appl.*, vol. 9, no. 4, pp. 38–44, Nov. 2010.
[43] P. L. Rosin and X. Sun, "Edge detection using cellular automata," in *Cellular Automata in Image Processing and Geometry*. Cham, Switzerland: Springer, 2014, pp. 85–103.
[44] M. Diwakar, P. K. Patel, and K. Gupta, "Cellular automata based edge-detection for brain tumor," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Aug. 2013, pp. 53–59.
[45] D. Tourtounis, N. Mitianoudis, and G. Ch. Sirakoulis, "Salt-n-pepper noise filtering using cellular automata," 2017, *arXiv:1708.05019*. [Online]. Available: http://arxiv.org/abs/1708.05019
[46] F. Qadir and I. Q. Shoosha, "Cellular automata-based efficient method for the removal of high-density impulsive noise from digital images," *Int. J. Inf. Technol.*, vol. 10, no. 4, pp. 529–536, Dec. 2018.
[47] B. Priego, A. Prieto, R. J. Duro, and J. Chanussot, "A cellular automata-based filtering approach to multi-temporal image denoising," *Expert Syst.*, vol. 35, no. 2, Apr. 2018, Art. no. e12235.
[48] Z. Zhang, Z. Wu, Q. Jiang, L. Du, and L. Hu, "Co-saliency detection based on superpixel matching and cellular automata," *TIIS*, vol. 11, no. 5, pp. 2576–2589, 2017.
[49] Y. Qin, M. Feng, H. Lu, and G. W. Cottrell, "Hierarchical cellular automata for visual saliency," *Int. J. Comput. Vis.*, vol. 126, no. 7, pp. 751–770, Jul. 2018.
[50] Y. Liu and P. Yuan, "Saliency detection using global and local information under multilayer cellular automata," *IEEE Access*, vol. 7, pp. 72736–72748, 2019.
[51] C. Sompong and S. Wongthanavasu, "An efficient brain tumor segmentation based on cellular automata and improved tumor-cut algorithm," *Expert Syst. Appl.*, vol. 72, pp. 231–244, Apr. 2017.
[52] Y. Liu, Y. Chen, B. Han, Y. Zhang, X. Zhang, and Y. Su, "Fully automatic breast ultrasound image segmentation based on fuzzy cellular automata framework," *Biomed. Signal Process. Control*, vol. 40, pp. 433–442, Feb. 2018.
[53] C. Li, L. Liu, X. Sun, J. Zhao, and J. Yin, "Image segmentation based on fuzzy clustering with cellular automata and features weighting," *EURASIP J. Image Video Process.*, vol. 2019, no. 1, pp. 1–11, Dec. 2019.
[54] G. Olague, F. Fernández, C. B. Pérez, and E. Lutton, "The infection algorithm: An artificial epidemic approach for dense stereo correspondence," *Artif. Life*, vol. 12, no. 4, pp. 593–615, Oct. 2006.
[55] B. Jana, P. Pal, and J. Bhaumik, "New image noise reduction schemes based on cellular automata," *Int. J. Soft Comput. Eng.*, vol. 2, no. 2, pp. 98–103, 2012.
[56] A. Wali and M. Saeed, "Biologically inspired cellular automata learning and prediction model for handwritten pattern recognition," *Biologically Inspired Cognit. Archit.*, vol. 24, pp. 77–86, Apr. 2018.
[57] S. Nichele and A. Molund, "Deep learning with cellular automaton-based reservoir computing," *Complex Syst.*, vol. 26, 2017, doi: 10.25088/ComplexSystems.26.4.319.
[58] N. H. Packard and S. Wolfram, "Two-dimensional cellular automata," *J. Stat. Phys.*, vol. 38, nos. 5–6, pp. 901–946, 1985.
[59] A. R. Khan, P. P. Choudhury, K. Dihidar, S. Mitra, and P. Sarkar, "VLSI architecture of a cellular automata machine," *Comput. Math. with Appl.*, vol. 33, no. 5, pp. 79–94, Mar. 1997.
[60] S. Uguz, H. Akin, I. Siap, and U. Sahin, "On the irreversibility of Moore cellular automata over the ternary field and image application," *Appl. Math. Model.*, vol. 40, nos. 17–18, pp. 8017–8032, Sep. 2016.
[61] S. Wang, Y. Tian, T. Pu, P. Wang, and P. Perner, "A hazy image database with analysis of the frequency magnitude," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 5, May 2018, Art. no. 1854012.
[62] J.-P. Tarel, N. Hautiere, A. Cord, D. Gruyer, and H. Halmaoui, "Improved visibility of road scene images under heterogeneous fog," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2010, pp. 478–485.
[63] J.-P. Tarel, N. Hautiere, L. Caraffa, A. Cord, H. Halmaoui, and D. Gruyer, "Vision enhancement in homogeneous and heterogeneous fog," *IEEE Intell. Transport. Syst. Mag.*, vol. 4, no. 2, pp. 6–20, Summer 2012.
[64] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[65] N. Hautière, J.-P. Tarel, D. Aubert, and É. Dumont, "Blind contrast enhancement assessment by gradient ratioing at visible edges," *Image Anal. Stereol. J.*, vol. 27, no. 2, pp. 87–95, Jun. 2008.
[66] T. M. Bui and W. Kim, "Single image dehazing using color ellipsoid prior," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 999–1009, Feb. 2018.

SURASAK TANGSAKUL received the B.Ed. degree in electronics and computer science from the King Mongkut's Institute of Technology, Ladkrabang, Thailand, in 2001, and the M.S. degree in computer science from Khon Kaen University, Thailand, in 2006, where he is currently pursuing the Ph.D. degree with the Machine Learning and Intelligence System Laboratory (MLIS), Department of Computer Science. His research interests include computer vision, pattern recognition, cellular automata, and machine learning.

SARTRA WONGTHANAVASU (Member, IEEE) received the B.S. degree in mathematics from Khon Kaen University, Thailand, in 1982, the M.S. degree in computer science from the Illinois Institute of Technology, USA, in 1996, and the Ph.D. degree in computer science from the Asian Institute of Technology, Thailand, in 2001. He has authored a variety of articles and also served as a reviewer of dozens of journals, such as the IEEE INTELLIGENT SYSTEMS, the IEEE TRANSACTIONS ON CYBERNETICS, *Expert Systems With Applications*, *Neural Computing and Applications*, and so on. His research interests include machine learning, computer vision, cellular automata, and knowledge engineering.