

An Online Application for Defogging of Videos and Images Based on RNN

Shamna P¹ Unnikrishnan S Kumar²

²Assistant Professor

^{1,2}MCET, India

Abstract— Most of the time climate decides the quality of an image. so, the situation of fog and haze make the image degraded. that’s why there is a need of de fogging of an image. The de-fogging system will favor the normal operations of the information system in the fields of military, transportation and safety monitoring. for the de-fogging purpose we are using a recognition algorithm based on recurrent neural network. . At present, the mainstream image de-fogging algorithm mainly uses a variety of fog related color features, however, different color prior knowledge often has its own scene limitation. We use sparse automatic coding machine to extract the texture features of the image, and extract all kinds of fog related color features. Then, we use the recurrent neural network to implement sample training process, and we obtain the mapping relationship between texture structure features and color features and scene depth, and then we estimate the scene deep map of fog images. Finally, the atmospheric scattering model is used to recover the fog free image according to the scene deep map. Experiments show that the proposed algorithm can effectively obtain the scene depth of the image, and recover the ideal fog free image. Also, we are giving an interface with telegram application.

Keywords: Recurrent Neural Network, Video Image, De-Fogging Recognition Algorithm, Network Structure

I. INTRODUCTION

In the bad weather condition like fog and haze there will be some water droplets and some dust particles in the atmosphere. as a result of this scattering or absorption of light will happen. it affect the image quality badly. so there is a great significance to implement a de fogging system. [7], [8].

Image defogging methods are mainly divided into two categories. First one is based on image enhancement method and second one is based on physical method. In image enhancement method defogging is achieved by enhancing the contrast of the image to achieve the purpose of de fogging. This method can be applied and then improve the existing mature image processing algorithm, and haze image can get better fog removal effect. Physical method is based on physical model, this method studies the objective mechanism of some atmospheric fog and image degradation, builds the atmospheric scattering model, and then recovers the fog image according to the physical model.

Artificial neural network is a branch of machine learning field. Deep learning stems from the artificial neural network, which can better simulate the brain structure, realize the abstraction process of cognitive process, and solve the problem of insufficient depth. There are many branches of deep learning, and the current research focuses are the convolutional neural network and recurrent neural network. RNN is a network system with strong learning ability. It can process several related information, which is suitable for processing time series data, such as speech recognition, text

generation, machine translation, sequence prediction, etc.. In order to calculate the error value of each layer network, RNN usually uses the back propagation through time algorithm (BPTT) [9]. However, BPTT cannot solve the problem of long time dependence, so the algorithm will bring about gradient disappearance and gradient explosion. In order to solve the problem of gradient vanishing and gradient explosion, we optimize the learning algorithm and configuration network.

II. RECURRENT NEURAL NETWORK

A. RNN

The network structure of RNN includes input layer, hidden layer and output layer. In theory, RNN can learn the knowledge of arbitrary length sequences. However, as intervals increase, it becomes difficult for RNN to learn the relationship between the connections, resulting in a time dependent problem.

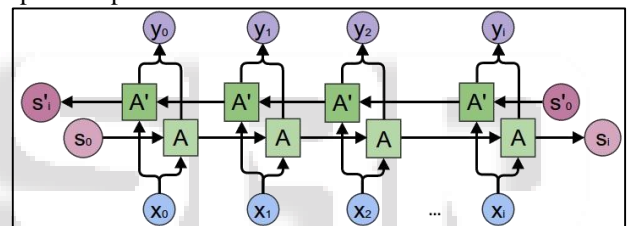


Fig. 1: Structure of LSTM.

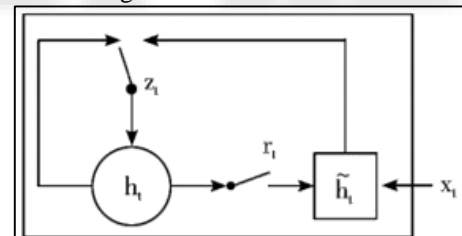


Fig. 2: Structure of GRU.

LSTM structure can learn long time dependence knowledge, and its core idea is memory cell and gate mechanism, as shown in Fig. 1. Memory cells run throughout the chain, with only a small amount of interaction in the middle, which ensures smooth and stable flow of information. The gate mechanism selectively allows information to pass through, adding or deleting information to neurons [16], [17]. The forgetting gate of LSTM is expressed as follows:

$$f_t = s W(f \cdot [h_{t-1}, x_t] + b_f)$$

The input gate of LSTM is expressed as follows:

$$i_t = s W(i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

The cellular renewal expression of LSTM is as follows:

$$c_t = f_t * c_{t-1} + i_t * c_t \quad (3)$$

The output gate of LSTM is expressed as follows:

$$o_t = s W(o \cdot [h_{t-1}, x_t] + b_o) \quad (4)$$

$$h_t = o_t * \tanh(c_t) \quad (5)$$

Inspired by the LSTM model gate mechanism, the GRU structure combines the input and the forgotten gates of

the LSTM structure into a single update gate, and merges the memory cell and the implicit state, as shown in Fig. 2. The update gate expression of GRU is as follows:

$$z_t = s(W_z \cdot [h_{t-1}, x_t]) \quad (6)$$

The reset gate expression of GRU is as follows:

$$r_t = s(W_r \cdot [h_{t-1}, x_t]) \quad (7)$$

The output gate of GRU is expressed as follows:

$$h_t = \tanh(W \cdot [h_{t-1} * r, x_t]) \quad (8)$$

$$h_t = -(1 - z_t) * h_{t-1} + z_t h_t \quad (9)$$

The expressions of the ReLU function and the softplus function are respectively:

$$f(x) = \max(0, x) \quad (10)$$

$$f(x) = \ln(1 + e^x) \quad (11)$$

In this article, we use the ReLU activation function and the softplus activation function in RNN. Compared with the traditional tanh function and the sigmoid function as shown Fig. 3, our method can deal with the gradient vanishing problem [18]-[20].

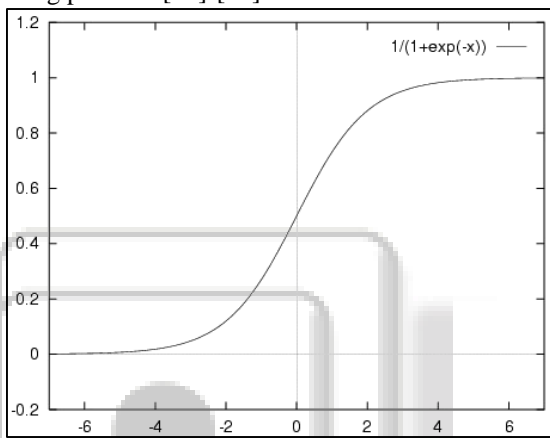


Fig. 3: Sigmoid function.

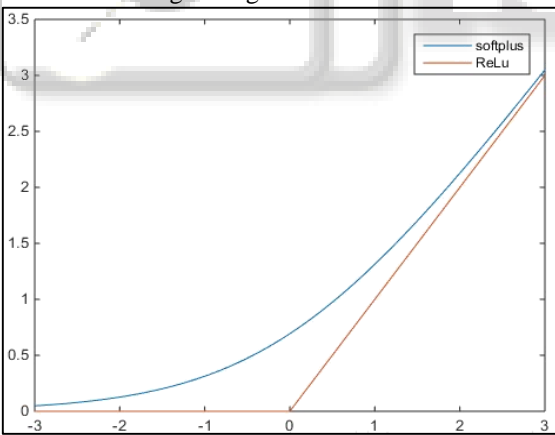


Fig. 4: ReLU and softplus function.

Because in the process of back propagation, the propagation of error information is closely related to the derivative of activation function and the simplified expression of error propagation is as follows:

$$E = \partial \cdot E \cdot f_{net}'(1) \cdot \dots \cdot W_1 \cdot f_{net}'(2) \cdot \dots \cdot W_n \cdot f_{net}'(n) \quad (12)$$

As shown in Fig. 4, for network learning, softplus function will provide more space for the network, and improve the learning ability of the model.

In the context of artificial neural networks, the rectifier is an activation function defined as (10) where x is the input to a neuron. This activation function was first introduced to a dynamical network by Hahnloser et al. in a

2000 paper in Nature. It has been used in convolutional networks more effectively than the widely used logistic sigmoid (which is inspired by probability theory) and its more practical counterpart, the hyperbolic tangent [21].

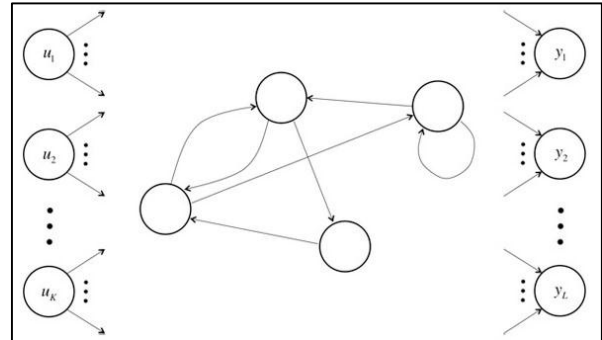


Fig. 5: Directed cyclic structure of RNN.

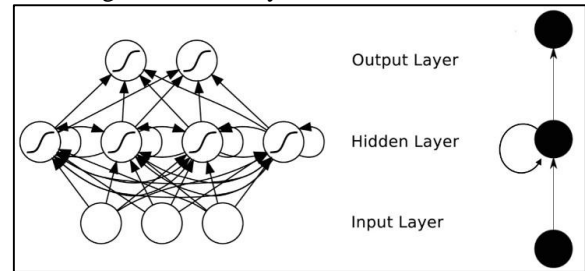


Fig. 6: Typical structure of RNNs.

B. Noisy ReLU and Leaky ReLU

Noisy ReLUs: Rectified linear units can be extended to include Gaussian noise, making them noisy ReLUs, giving: $f(x) = \max(0, x + Y)$, with $Y \sim N(0, \sigma(x))$. Noisy ReLUs have been used with some success in restricted Boltzmann machines for computer vision tasks.

Leaky ReLUs: allow a small, non-zero gradient when the unit is not active:

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

$$f(x) = \begin{cases} x, & x > 0 \\ 0.01x, & \text{elsewhere} \end{cases} \quad (13)$$

$$f(x) = \begin{cases} x, & x > 0 \\ ax, & \text{elsewhere} \end{cases}$$

Parametric ReLUs take this idea further by making the coefficient of leakage into a parameter that is learned along with the other neural network parameters:

$$f(x) = \begin{cases} x, & x > 0 \\ ax, & \text{elsewhere} \end{cases} \quad (14)$$

$$f(x) = \begin{cases} x, & x \geq 0 \\ a e^{-x}, & \text{elsewhere} \end{cases}$$

Note that for $a \leq 1$, this is equivalent to: $f(x) = \max(x, ax)$, and thus has a relation to "maxout" networks.

ELUs: Exponential linear units try to make the mean activations closer to zero which speeds up learning. It has been shown that ELUs can obtain higher classification accuracy than ReLUs:

$$f(x) = \begin{cases} x, & x \geq 0 \\ a e^{-x}, & \text{elsewhere} \end{cases}$$

$$f(x) = \begin{cases} x, & x \geq 0 \\ a e^{-x}, & \text{elsewhere} \end{cases} \quad (15)$$

where a is a hyper-parameter to be tuned and $a \geq 0$ is a constraint.

C. Directed Cyclic Structure

Unlike traditional FNN, RNN introduces directed loops, which can deal with problems of related information. The directional loop structure is shown in Fig. 5.

RNN is used to process sequence data. In the traditional neural network model, the nodes between each layer are connectionless. But this common neural network is powerless to solve many problems. RNN can record the previous information and apply it to the calculation of the current output.

In theory, RNN can handle any length of sequence data. But in practice, in order to reduce complexity, we usually assume that the current state is only related to the previous states. The typical structure of RNNs is shown in Fig. 6.

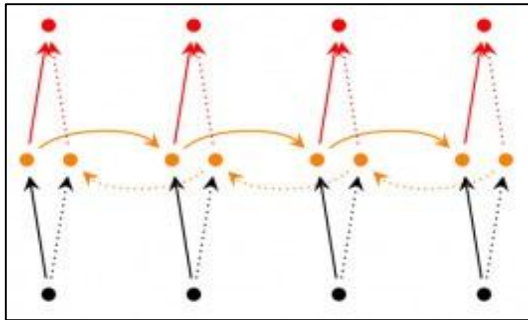


Fig. 7: Structure of Bidirectional RNNs.

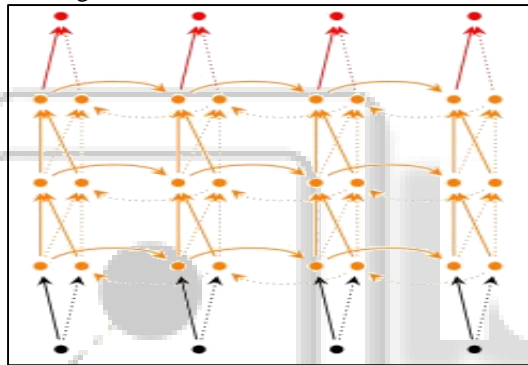


Fig. 8: Structure of Deep RNNs.

D. Improvement of RNNs

Bidirectional RNNs: Bidirectional RNN assumes that the current output is related to the previous sequence, and is also related to the subsequent sequence. Bidirectional RNN is made up of two RNNs. The output of Bidirectional RNN is determined by the state of the hidden layers of these two RNNs, as shown in Fig. 7.

Deep RNNs: Deep RNNs is similar to Bidirectional RNNs, and the input of each step includes multilayer networks, as shown in Fig. 8. In this way, the network has more powerful expression and learning ability, but the complexity is also improved, and more training data is needed.

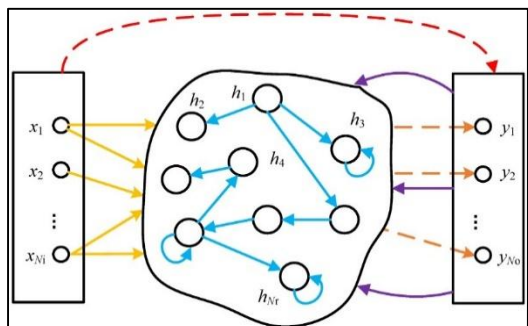


Fig. 9: The structure of ESN.

E. Echo State Networks

Although ESN is also a kind of RNN, it is very different from the traditional RNN. ESNs has three characteristics, as follows.

- 1) The core structure of ESNs is a reserve pool that is randomly generated and remains unchanged. The reserve pool is a large, randomly generated, and sparsely connected cyclic structure.
- 2) The weight matrix of the reserve pool is the only part that needs to be adjusted.
- 3) A simple linear regression can complete the network training process.

Structurally, ESN is a special type of recurrent neural network. The basic idea is: replacing the intermediate layer in the classical neural network with the large scale random connection loop network, so as to simplify the training process of the network. Therefore, the key to ESN is the intermediate reserve pool.

The parameters in the network include the connection weight matrix of the nodes in the reserve pool, the connection weight matrix between the input layer and the reserve pool, the feedback connection weight matrix between the output layer and the reserve pool, and the connection weight matrix between the input layer, the reserve pool and the output layer to the output layer. The structure of ESN is shown in Fig. 9.

F. Gated Recurrent Unit RNN

GRU is also an improved version of RNN, including the following improvement.

- 1) The data at different locations in the sequence have different effects on the current state of the hidden layer.
- 2) The error may be caused by one or several data, so it is necessary to update the corresponding weights only. The structure of GRU is shown in Fig. 10.

$$\text{where } z_t = \sigma(W z^z_t + U h^z_{t-1}) \tag{16}$$

$$r_t = \sigma(W x^r_t + U h^r_{t-1}) \tag{17}$$

$$\tilde{h}_t = z_t \# h^*_{t-1} + (1 - z_t) * \tilde{h}_t \tag{19}$$

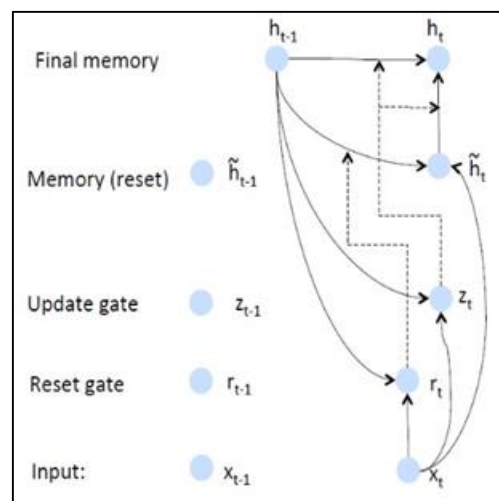


Fig. 10: The structure of GRU-RNN.

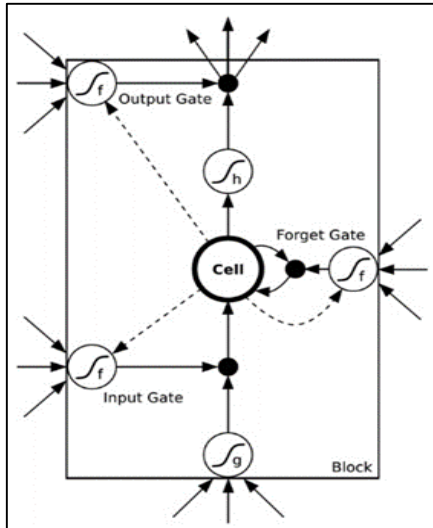


Fig. 11: The network structure of LSTM.

G. LSTM Network

LSTMs is similar to GRUs, which is very popular now. It is not essentially different from the general RNNs structure, but it uses different functions to calculate the state of the hidden layer.

The network structure is very effective for long sequence dependence problems. The network structure of LSTM is shown in Fig. 11.

The problem that LSTMs solves is also mentioned in GRU. There are also some differences between LSTM and GRU as follows.

- 1) The calculation methods of new memory are calculated according to the previous state and input, nevertheless, these is a reset gate which controls previous state in GRU, while there is no gates in LSTMs.
- 2) These two structures have different ways to generate new states. LSTMs has two different gate, namely, forget gate (f gate) and input gate (I gate), while GRUs has only one update gate (Z gate).
- 3) LSTM can adjust the size of state, but GRU has no regulation mode.

H. ClockWork Rnn(CW-RNN)

CW-RNN is a RNN which driven by clock frequency. It divides the hidden layer into several blocks and each group processes the input according to its own prescribed clock frequency. In order to reduce the complexity of the standard RNN, CW-RNN reduces the number of parameters, improves the network performance, and accelerates the training process of the network.

CW-RNN uses different clock frequencies to solve long time dependency problems. Therefore, all hidden layer groups do not work at each step, which speeds up the training of the network. CW-RNN is similar to the SRN's network structure, including input layer, hidden layer, and output layer, and there are also connections between layers.

The network structure of CW-RNN is shown in Fig. 12.

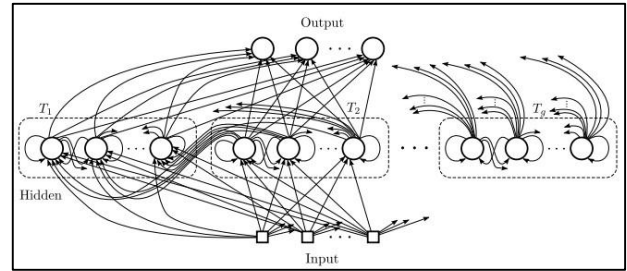


Fig. 12: The structure of CW-RNN.

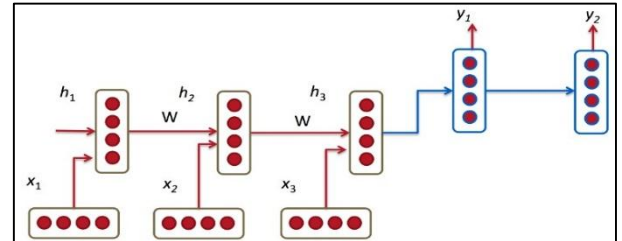


Fig. 13: Performances of RNN, LSTM and CW-RNN.

I. Evaluation Results of different RNNs

In CW-RNNs, slow groups (large cycles) handle, retain, and output long dependent information, while fast groups update. The error propagation of CW-RNN is similar to that of traditional RNN, but the error is propagated only in the hidden layer group, and the hidden layer group of the non-executing state also copies the backward propagation of the previous hidden layer group.

In this section, we evaluate RNN, LSTM, CW-RNN in order to better evaluate the performances of different models.

As shown in Fig. 13, the green solid line is the predicted result, and the blue scatter is the real result. Each model learns from the first half, and then predicts the second half.

LSTM model is similar to average moving process, but CW-RNN has better performance. The input layer, hidden layer and output layer of the three models have the same number of nodes, and each of them only contains one hidden layer; the weights follow the Gauss distribution whose mean value is 0 and the standard deviation is 0.1; the initial state of hidden layer is 0.

III. TELEGRAM APPLICATION

Telegram Messenger is a messaging app that works over the internet, just like WhatsApp or Facebook Messenger. That means you can send messages for free by using a wi-fi connection or your mobile data allowance (providing you have enough data).

Telegram's main selling point is security. It claims all its activities – including chats, groups and media – are encrypted, meaning even if they are intercepted, they won't be visible without being deciphered first. However, some security experts have cast doubt on how secure Telegram is.

Telegram can be used on smartphones, tablets, laptops and desktop computers. The Telegram app is available for Android, iOS, Windows Phone, Windows NT, macOS and Linux.

Telegram is a cloud-based instant messaging and voice over IP service developed by Telegram Messenger

LLP, a privately held company registered in London, United Kingdom, founded by the Russian entrepreneur Pavel Durov and his brother Nikolai. Telegram client apps are available for Android, iOS, Windows Phone, Windows NT, macOS and Linux. Users can send messages and exchange photos, videos, stickers, audio and files of any type.

Telegram's client-side code is open-source software but the source code for recent versions is not always immediately published, whereas its server-side code is closed-source and proprietary. The service also provides APIs to independent developers. In March 2018, Telegram stated that it had 200 million monthly active users. According to its CEO, as of April 2017, Telegram's annual growth rate was greater than 50%.

Messages and media in Telegram are only client-server encrypted and stored on the servers by default. The service provides end-to-end encryption for voice calls, and optional end-to-end encrypted "secret" chats between two online users, yet not for groups or channels.

Telegram's security model has received notable criticism by cryptography experts. They criticized the general security model of permanently storing all contacts, messages and media together with their decryption keys on its servers by default and by not enabling end-to-end encryption for messages by default. Pavel Durov has argued that this is because it helps to avoid third-party unsecure backups, and to allow users to access messages and files from any device.] Cryptography experts have furthermore criticized Telegram's use of a custom-designed encryption protocol that has not been proven reliable and secure.

Telegram has faced censorship or outright bans in some countries over accusations that the app's services have been used to facilitate illegal activities, such as protests and terrorism, as well as declining demands to facilitate government access to user data and communications.

A. Bots

In June 2015, Telegram launched a platform for third-party developers to create bots. Bots are Telegram accounts operated by programs. They can respond to messages or mentions, can be invited into groups and can be integrated into other programs. It also accepts online payments with credit cards and Apple Pay. Dutch website Tweakers reported that an invited bot can potentially read all group messages when the bot controller changes the access settings silently at a later point in time. Telegram pointed out that it considered implementing a feature that would announce such a status change within the relevant group.[66] Also there are inline bots, which can be used from any chat screen. In order to activate an inline bot, user needs to type in the message field a bot's username and query. The bot then will offer its content. User can choose from that content and send it within a chat.

IV. LITERATURE REVIEW

Video Deblurring Algorithm Using Accurate Blur Kernel Estimation and Residual Deconvolution Based on a Blurred-Unblurred Frame Pair Dong-Bok Lee, Shin-Cheol Jeong, Yun-Gu Lee, and Byung Cheol Song

this paper presents a novel motion deblurring algorithm in which a blurred frame can be reconstructed

utilizing the high-resolution information of adjacent unblurred frames. First, a motion-compensated predictor for the blurred frame is derived from its neighboring unblurred frame via specific motion estimation. Then, an accurate blur kernel, which is difficult to directly obtain from the blurred frame itself, is computed using both the predictor and the blurred frame. Next, a residual deconvolution is applied to both of those frames in order to reduce the ringing artifacts inherently caused by conventional deconvolution. The blur kernel estimation and deconvolution processes are iteratively performed for the deblurred frame. Simulation results show that the proposed algorithm provides superior deblurring results over conventional deblurring algorithms while preserving details and reducing ringing artifacts.

Multi-Scale Patch-Based Image Restoration Vardan Papyan, and Michael Elad, Fellow, IEEE

The core idea is to decompose the target image into fully overlapping patches, restore each of them separately, and then merge the results by a plain averaging. This concept has been demonstrated to be highly effective, leading often times to state-of-the-art results in denoising, inpainting, deblurring, segmentation, and other applications. While the above is indeed effective, this approach has one major flaw: the prior is imposed on intermediate (patch) results, rather than on the final outcome, and this is typically manifested by visual artifacts. The EPLL (Expected Patch Log Likelihood) method by Zoran and Weiss was conceived for addressing this very problem. Their algorithm imposes the prior on patches of the final image, which in turn leads to an iterative restoration of diminishing effect. In this work we propose to further extend and improve the EPLL by considering a multi-scale prior. Our algorithm imposes the very same prior on different scale patches extracted from the target image. While all the treated patches are of the same size, their footprint in the destination image varies due to subsampling

Fast single image fog removal using wiener filter method Kristofor B. Gibson and Truong Q. Nguyen

in this paper a fast single image defogging method that uses a novel approach to refining the estimate of amount of fog in an image with the Locally Adaptive Wiener Filter. We provide a solution for estimating noise parameters for the filter when the observation and noise are correlated by decorrelating with a naively estimated defogged image. We demonstrate our method is 50 to 100 times faster than existing fast single image defogging methods and that our proposed method subjectively performs as well as the Spectral Matting smoothed Dark Channel Prior method.

An overview on defogging a fogged image using histogram equalization. Garima Kadian, Dr. Rajiv Kumar In this paper, reported algorithms for the removal of fog are reviewed. Fog reduces the visibility of scene and thus performance of various computer vision algorithms which use feature information. Formation of fog is the function of the depth. Estimation of depth information is under constraint problem if single image is available. Hence, removal of fog requires assumptions or prior information. Fog removal algorithms estimate the depth information with various assumptions, which are discussed in detail here. Fog removal algorithm has a wide application in tracking and navigation, consumer electronics, and entertainment industries.

Review of video and image defogging algorithms and related studies on image restoration and enhancement YONG XU, JIE WEN, LUNKE FEI, AND ZHENG ZHANG

Video and images acquired by a visual system are seriously degraded under hazy and foggy weather, which will affect the detection, tracking, and recognition of targets. Thus, restoring the true scene from such a foggy video or image is of significance. The main goal of this paper was to summarize current video and image defogging algorithms. We first presented a review of the detection and classification method of a foggy image. Then, we summarized existing image defogging algorithms, including image restoration algorithms, image contrast enhancement algorithms, and fusion-based defogging algorithms. We also presented current video defogging algorithms. We summarized objective image quality assessment methods that have been widely used for the comparison of different defogging algorithms, followed by an experimental comparison of various classical image defogging algorithms. Finally, we presented the problems of video and image defogging.

Markov Random Field Model for Single Image Defogging

Laurent Caraffa, Jean-Philippe Tarel.

In this paper, we propose a novel MRF model of the single image defogging problem which applies to all kinds of images but can also easily be refined to obtain better results on road images using the planar constraint.

V. EXISTING SYSTEM

To meet the needs of real-time processing of existing video's de-fogging process, a recognition algorithm based on recurrent neural network is proposed in this paper, and experiments show that the proposed algorithm can effectively obtain the scene depth of the image, and recover the ideal fog free image. There are two main technical difficulties in the image de-fogging method. (1) Atmospheric light is estimated to be then easily disturbed by bright areas such as the sky while leading to the overvaluation. (2) The calculation of transmittance is easy to produce unreasonable texture fluctuation, which leads to the reduction of its reliability.

The guide filter is a novel edge preserving smoothing filter, which can effectively smooth the background details and maintain the edge features of the image scene without gradient reversal effect. The guiding filter has the edge preserving property because the guiding image and the output image satisfy the linear relation in the local window, so that the gradient of the two is similar. Image restoration processing is based on the analysis of the degradation mechanism of the haze image, the use of prior knowledge or the assumption of image degradation, the establishment of a physical model of image degradation or estimation of the properties of the fog, so as to achieve targeted defogging and scene restoration [10]-[12]. This lifting scheme based on wavelet transform and the traditional wavelet biggest difference is to give up the idea of binary stretching and translation completely out of the Fourier analysis [13].

This method not only inherits the multi-scale resolution of the first-generation wavelet, but also has the

characteristics of fast computation [8], [9]. This will be considered for our model.

VI. DE-FOGGING ALGORITHM OF VIDEO IMAGE

A. Atmospheric Scattering Model

Influenced by fog, haze and other environmental factors, the impurities in the air will affect the direct observation of human vision, which also blur the definition of the scene. Therefore, the effective restoration of scene clarity plays an important role in image processing and pattern recognition. The atmospheric scattering model is mainly composed of the incident light attenuation model and the atmospheric optical imaging model.

$$I(x) = J(x) \cdot t(x) + A(1-t(x)) \quad (20)$$

where $I(x)$ is the actual observed fog image; $J(x)$ is the processed image; A represents the global atmospheric light; $t(x)$ is the transmittance of medium; $J(x) \cdot t(x)$ represents the attenuation model of incident light; $A(1-t(x))$ represents the atmospheric optical imaging model.

B. Fog Restoration Model

The observed fog images are known. In order to recover the clear fog free images, we replace some parameters in above equations. Thus, we have:

$$t(x) = -1 \cdot T(x) \quad (21)$$

$$I(x) = J(x)(1 - T(x)) + T(x) \quad (22)$$

We can reconstruct the fog free clear image by substituting the parameters of transmittance and global atmospheric light into fog image restoration model.

C. Color Space Conversion Module

In video image display, the general display does not support the YCrCb format, and the YCbCr4:2:2 data format is obtained by ADV7180 decoding.

We need to convert YUV4:2:2 to YCbCr4:4:4, and we obtain video data of YCbCr4:4:4, where Y, Cb and Cr are respectively 8-bits data.

We assign Cb and Cr to the next two image pixels. Then, the YCrCb format video image data is converted into the format data of RGB color space. The conversion formulas of YCrCb and RGB are as follows:

$$R = 1.164(Y - 16) + 1.596(Cr - 128) \quad (23)$$

$$G = 1.164(Y - 16) - 0.813(Cr - 128) + 0.391(Cb - 128) \quad (24)$$

$$B = 1.164(Y - 16) + 2.018(Cb - 128) \quad (25)$$

After decoding processing, Y, Cb and Cr are 8 bit data. Because the color component ADV7123 requires 10 bits data, so we need to shrink both sides of the formula of 128 times at the same time.

D. Image storage module

The main function of SDRAM (Synchronous Dynamic Random Access Memory) is to cache the odd and even field signals in the video signal, and read the cached video data, and then output the video signal to form a complete frame image. The SDRAM control module mainly controls the initialization of SDRAM, read-write control, automatic refresh, etc.

After the SDRMA initialization process, we can do read and write operations on SDRMA. We design a control module to drive the SDRAM memory. The driver module is

mainly divided into the controller module and the data flow module. The controller module completes the initialization of power on, automatic refresh, read and write control, etc.

VII. PROPOSED SYSTEM

Here we are proposing a new improved video dehazing system in which each frames dehaze more accurately and efficiently. here video is considered as a file and each file is composed of many frames. Each frames are dehazed separately based on atmospheric scattering model and fog restoration model. for this we use some equations which are described in existing system.to restore the fog we have to calculate some features like contrast, correlation, energy, homogeneity, standard deviation , mean, skewness and kurtosis which are known as color and texture features. from these values we can evaluate atmospheric light, transmission estimate and dark channel values. then after comparing these values the hazed portions can be identified and dehazing can be done more accurately. these dehazing processes can be repeated for every frames.

The input can be directly read from the system itself and also it provides an online application bot to interact with this system. we use telegram android application for this purpose.in telegram we create a group and adding contacts. The bot program is executed by matlab which checks latest messages arriving in the telegram group automatically. The bot processes the video file, and send back the dehazed version of the file back to the group itself. Any member in the group can be send hazed images or videos to this group and it can be read by the bot, and get back the dehazed image or video in a short time. This makes it easy to dehaze the video or image from anywhere in the world without having physical access to a computer.

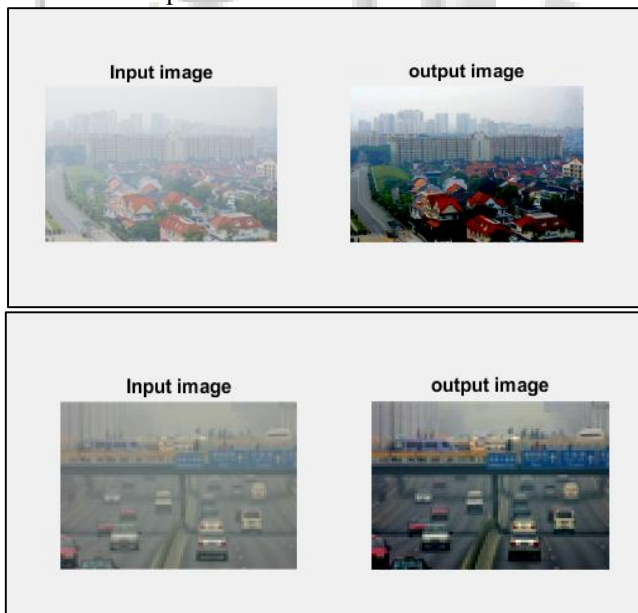


Fig. 15: Input and output images

REFERENCES

[1] Video Deblurring Algorithm Using Accurate Blur Kernel Estimation and Residual Deconvolution Based on a Blurred-Unblurred Frame Pair Dong-Bok Lee, Shin-Cheol Jeong, Yun-Gu Lee, and Byung Cheol Song

[2] -Scale Patch-Based Image Restoration Vardan Papyan, and Michael Elad, Fellow, IEEE

[3] Fast single image fog removal using wiener filter method Kristofor B. Gibson and Truong Q. Nguyen

[4] An overview on defogging a fogged image using histogram equalization. Garima Kadian, Dr. Rajiv Kumar

[5] An overview on defogging a fogged image using histogram equalization. Garima Kadian, Dr. Rajiv Kumar

[6] Markov Random Field Model for Single Image Defogging Laurent Caraffa. Jean-Philippe Tarel

[7] Z. W. He and W. Chen, "Fast Wiener-filter based dehazing method for vision," Video Engineering, vol. 39, no. 15, pp. 15-18, 2015.

[8] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," arXiv preprint, arXiv:1703.02910, 2017.

[9] Y. LeCun, Y. Bengio, G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, Feb. 2015.

[10] C. Wang, H. Yang, C. Bartz and C. Meinel, "Image captioning with deep bidirectional LSTMs," ACM on Multimedia Conference, Oct. 2016, pp. 988-997.

[11] S. Zhang, H. Wang, and W. Huang, "Two-stage plant species recognition by local mean clustering and weighted sparse representation classification," Cluster Computing, vol. 20, no. 2, pp.1517-1525, 2017.

[12] C. Wang, H. Yang, and C. Meinel, "A deep semantic framework for multimodal representation learning," Multimedia Tools and Applications, vol. 75, no. 15, pp. 9255-9276, Aug. 2016.

[13] S. Zhang, H. Wang, W. Huang, and Z. You, "Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG," Optik-International Journal for Light and Electron Optics, vol. 157, pp. 866-872, Nov. 2017.

[14] C. Wang, H. Yang and C. Meinel, "Deep semantic mapping for cross-modal retrieval," In proc. ICTAI, 2015, pp. 234-241.

[15] C. Wang, "RRA: Recurrent residual attention for sequence learning," arXiv preprint, arXiv:1709.03714, 2017.

[16] M. Zareapoor, P. Shamsolmoali, D.K. Jain, H. Wang, and J. Yang, "Kernelized support vector machine with deep learning: an efficient approach for extreme multiclass dataset," Pattern Recognition Letters, DOI: 10.1016/j.patrec.2017.09.018, 2017.

[17] C. Wang, H. Yang, and C. Meinel, "Visual-textual late semantic fusion using deep neural network for document categorization," Neural Information Processing, LNCS. Springer, Cham, Nov. 2015, pp. 662-670.

[18] D. He, N. Kumar, S. Zeadally and H. Wang, "Certificateless provable data possession scheme for cloud-based smart grid data management systems," IEEE Transactions on Industrial Informatics, vol. PP, no. 99, pp: 1-1, Oct. 2017.

[19] M. A. Casteleiro, G. Demetriou, W. Read, M. J. Fernandez-Prieto, D. Maseda-Fernandez, G. Nenadic, J. Klein, J. Keane and R. Stevens, "Deep learning meets semantic web: a feasibility study with the cardiovascular disease ontology and PubMed citations," In ODLS, 2016, pp. 1-6.

- [20] H. Ravishankar, P. Sudhakar, R. Venkataramani, S. Thiruvankadam, P. Annangi, N. Babu, and V. Vaidya, "Understanding the mechanisms of deep transfer learning for medical images," arXiv preprint, arXiv:1704.06040, 2017.
- [21] K. Muhammad, R. Hamza, J. Ahmad, J. Lloret, H. H. G. Wang and S. W. Baik, "Secure surveillance framework for IoT systems using probabilistic image encryption," IEEE Transactions on Industrial Informatics, vol. PP, no. 99, pp: 1-1, Jan. 2018.

