https://doi.org/10.2352/issn.2694-118X.2020.LIM-12

CNN-based Rain Reduction in Street View Images

Simone Zini, Simone Bianco, Raimondo Schettini
Department of Informatics, Systems and Communication, University of Milano - Bicocca, viale Sarca, 336 Milano, Italy
{***s.zini1@campus.unimib.it, simone.bianco@unimib.it, raimondo.schettini@unimib.it***}

Abstract

Rain removal from pictures taken under bad weather conditions is a challenging task that aims to improve the overall quality and visibility of a scene. The enhanced images usually constitute the input for subsequent Computer Vision tasks such as detection and classification. In this paper, we present a Convolutional Neural Network, based on the Pix2Pix model, for rain streaks removal from images, with specific interest in evaluating the results of the processing operation with respect to the Optical Character Recognition (OCR) task. In particular, we present a way to generate a rainy version of the Street View Text Dataset (R-SVTD) for "text detection and recognition" evaluation in bad weather conditions. Experimental results on this dataset show that our model is able to outperform the state of the art in terms of two commonly used image quality metrics, and that it is capable to improve the performances of an OCR model to detect and recognise text in the wild.

Introduction

In the last years, low-level image processing has improved a lot with the introduction of Convolutional Neural Networks, permitting to outperform classical handcrafted methods in most tasks such as Super-Resolution, Image Denoising, Image Colorization, Image Dehazing and Deraining. Those methods are intended to enhance the input images that suffer from problems of different nature in order to improve the quality and visibility as perceived by humans or for subsequent automatic systems like automatic object detectors, etc.

In this work, we focus our attention on Image Deraining, where the objective is to remove rain from images taken during bad weather conditions, more specifically in situations where the visibility is occluded by rain streaks and haze. We are not considering the case in which we have raindrops over the camera lenses. In the last years, a lot of CNN based models for single image deraining have been presented [3–5, 13, 16–18]. Li et al. [9] presented a benchmark of all the current state of the art models for the deraining task, considering the different existing datasets and also the possibility to improve detectors' performances with different methods.

Inspired by this last work we managed to see the effect of this kind of processing on rainy street view images in order to improve the performance of text detectors and Optical Character Recognition (OCR) systems. In this work, we propose an autoencoder model, inspired by the Pix2Pix framework [6] and the one proposed by Zhang et al. [18], for the removal of rain streaks from rainy street view images. Subsequently, we evaluate the performances of the model using two classical image quality metrics and evaluate the results of an OCR model on the resulting derained images.

Proposed Method

Inspired by the results obtained by Convolutional Neural Networks and in particular GANs in low-level image processing tasks such as Super Resolution, Image Colorization, Image Inpainting, Noise Removal, etc... we decided to use a U-Net style architecture trained using a discriminative network in a conditional Generative Adversarial Network framework.

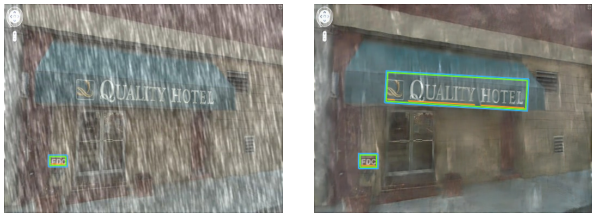


Figure 1: Text detection obtained using Google Cloud Vision API. For the rainy image only the small text has been detected while for the processed version the entire sign has been correctly recognized.

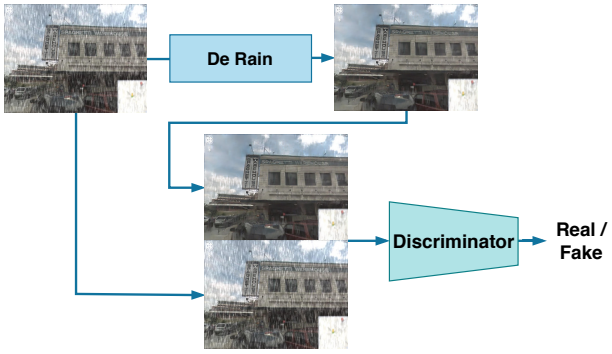


Figure 2: Training system with Conditional patchGAN.

Network Details

The structure of the DeRaining CNN is based on the U-Net [14] architecture, with the addition of skip connections as done for Pix2Pix network [6]. The architecture is shown in Figure 3.

Based also on recent works related to image generation we applied some changes to the classical U-Net architecture. First of all, as done in [10] and [12] we decided to remove the normalization layers from the model, in order to avoid the generation of artifacts. We substituted the MaxPooling operation with convolutions with stride 2 to reduce feature spatial dimensions without losing useful information for the restoration process. Lastly, to reduce artifacts coming from the application of the Deconvolutional Layers for the decoding part, we adopted a combination of Bilinear upsampling and 2D Convolutional layers.

In order to train the model in a GAN framework we adopted a patchGAN discriminative network, trained in a Conditional GAN training approach [11], using both generated and input images as input to the discriminator to better classify fake and real images, similarly to the discriminator used for Pix2Pix.

The architecture is shown in Figure 4 while a scheme of the entire training method is shown in Figure 2.

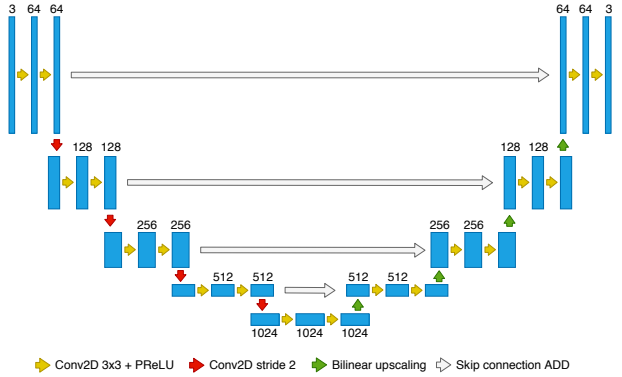


Figure 3: U-Net style architecture of the generative network. The max pooling layers have been replaced with convolutions with strides > 1 and the upscaling operation is performed with Bilinear Interpolation combined with convolutions.

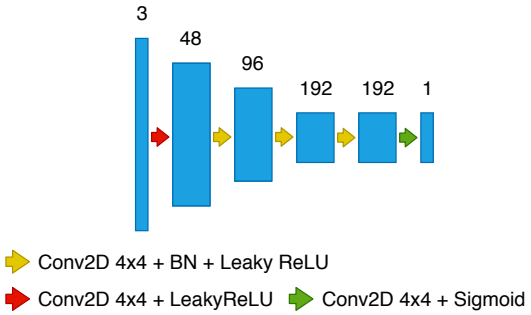


Figure 4: PatchGAN style discriminative network architecture.

Loss Function

The loss function used to train the model is defined as:

$$Loss = \lambda_e * L_e + \lambda_{adv} * L_{adv} + \lambda_p * L_p. \quad (1)$$

which is the combination of three loss functions, weighted by three different weight values $\lambda_e, \lambda_{adv}, \lambda_p$

Given an image pair $\{x, y\}$ with C channels, width W and height H (i.e. $C \times W \times H$), where x is the input image and y is the corresponding target, we define the three loss function as follows.

The per-pixel Euclidean loss, defined as:

$$L_e = \frac{1}{CWH} \sum_{c=1}^{C} \sum_{w=1}^{W} \sum_{h=1}^{H} ||\phi_E(x^{c,w,h}) - y^{c,w,h}||_2^2, \quad (2)$$

where $\phi_E(\cdot)$ is the learned network for rain removal.

The Perceptual loss [7] defined as distance function between features extracted from the target and output images, using the pre-trained VGG network:

$$L_p = \frac{1}{C_i W_i H_i} \sum_{c=1}^{C_i} \sum_{w=1}^{W_i} \sum_{h=1}^{H_i} ||\phi_E(x^{c,w,h}) - V(y_B^{c,w,h})||_2^2, \quad (3)$$

where $V(\cdot)$ represents a non-linear CNN transformation (VGG16 network).

Finally, the original *GAN* loss described as:

$$L_{adv} = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_x[\log(1 - D(x, G(x)))], \quad (4)$$

where $G(\cdot)$ is the trained generative network for image deraining.

Training Data

In order to train the model for the rain removal task, a dataset made of *input-target* images created with synthetic rain masks has been adopted. The dataset has been presented by Zhang et al. [18]: the entire dataset is made of 700 images, where 500 images have been randomly taken from the UCID dataset [15] and 200 images are randomly chosen from the BSD-500 training set [2]. The validation has been performed with the images from the test set made with 100 images, 50 from the UCID dataset and 50 from the BSD-500 dataset [2].

For each image, a rain mask has been chosen (from a set of 10 different ones) and applied. This operation has been manually done by Zhang et al. [18] using Photoshop. Moreover, in order to test the models with "real" rainy images, Zhang et al. collected a set of 50 natural images.

Since for the training phase the number of images is limited, we decided to use both image flipping and rotation in order to augment the dataset. All of the images have been cropped to a common size of 256×256, in the case in which the images were bigger, and upscaled to that dimension, in the case in which the images were smaller.

Training Details

The model has been written in PyTorch v1.3.1 and trained on an Nvidia Titan V GPU. The training has been done with batch size 8 for a total amount of 1K epochs. The model has been trained using Adam optimizer [8] with a starting learning rate of 10^{-5} for both generative and discriminative networks. For the balancing of the loss, in order to stabilize the training, we choose respectively $\lambda_e = 1$, $\lambda_p = 0.1$ and $\lambda_{adv} = 6.6 * 10^{-3}$.

Experimental Results

Rainy Street View Images Synthesizing

In order to test the capability of the processing operation to improve the results of OCR methods, we decided to use images of street scenes containing text areas. To this end, we decided to adopt the STREET VIEW TEXT DATASET [1], which contains 350 images taken from Google Street View with high variability in text from signs.

Since none of these images has been taken in bad weather conditions (such as rainy days, presence of haze or snow) we applied to each image a rain mask created similarly to [18]: instead of using Photoshop with a limited number of human-generated masks, we used MATLAB, creating for each image a new random mask. A set of parameters are randomly chosen in a range of possible values, empirically defined, in order to obtain the most realistic rainy images possible. The resulting dataset has been called RAINY STREET VIEW TEXT DATASET (R-SVTD). Some examples of the R-SVTD are shown in Figure 5.

Quality Comparison

The first comparison has been done using the most commonly used full reference image quality metrics, i.e. PSNR and SSIM. In Table 1 we compare our method with other four methods in the state of the art: Fu et al. CNN [4] and DDN [5], Yang et al. JORDER [16] and Zhang et al. [17].

As can be seen from the table, our method shows better results in terms of both the image quality metrics considered: with respect to the state of the art methods, we obtained an improvement of +1.5328 dB in terms of PSNR and +0.0027 in terms of SSIM, while with respect to the rainy input images we have an improvement of quality of +3.9642 dB and +0.0911 respectively for PSNR and SSIM.

Figure 5: Some images from the R-SVTD after the application of the random rain mask with MATLAB. To improve the quality of the images, the mask has been created by combining synthesized streaks and haze.

OCR test

Due to the lack of the possibility to make a quantitative comparison of the model in terms of accuracy in text detection and recognition, we decided to use the OCR system provided by Google Cloud Vision API for a visual comparison. In Figure 6 there are some images and their text detection results before and after the application of the proposed deraining method.

As can be seen from the examples reported, the proposed deraining method tends to improve the results of the OCR. In most of the cases, the OCR is able to detect text areas that were not detected before, even if the text recognition is not always completely correct. This improvement can be seen mainly in the case of *heavy rain* conditions while in general in the other cases the improvement is not that significant since the OCR used is capable to correctly detect the text area. In those cases, the proposed deraining method improves the recognition of few letters with respect to the rainy version. In the 42% of the cases, i.e. 147 out of 350 images from the RAINY STREET VIEW TEXT DATASET, the rain removal processing step improved the results in terms of both text detection and recognition.

Conclusions

In this work we proposed a Convolutional Neural Network, based on the Pix2Pix model, for image rain streaks removal. The proposed model has been compared on a dataset composed by street view scenes, to which we have added synthetically generated rain. This dataset has been called RAINY STREET VIEW TEXT DATASET (R-SVTD). Comparisons with the other state of the art methods shown that our model outperforms the previous obtained results in terms of PSNR and SSIM indexes, with a respective improvement of +1.5328 dB and +0.0027. Using the R-SVTD dataset we also showed how the model is capable to restore the structures of the degraded images in order to improve the results of an OCR model used after the restoration.

In the end, we obtained promising results that encourage us to continue working in this direction with a major focus on optimization of those methods, specifically for those kind of tasks limited by the nature of the images.

As future step, is necessary to put the attention on some points that we find out after those experiments were done. At the moment the model is trained for the reconstruction of general content images since the training has been performed on those kinds of contents. A first step can be related to the training of a model for the removal of rain in relation to the specific content or information in which we are interested to restore. A second point is related to the fact that not in all of the cases the processing operation gained some improvement. In some cases, the models tend to introduce artifacts. In these cases, the text that was originally well recognized, change for some letters because of the wrong enhancement during the processing. Putting attention on that fact, the next step will be related to the reduction of undesired artifacts in the enhancement operation. Another possible route to follow is the one that considers the use the results of one or more detectors as objective function for the training of the models, with the purpose to obtain CNNs to specifically improve the results related to the next step of detection and recognition.

Table 1: Comparison of the methods in terms of PSNR and SSIM indexes for the RAINY STREET VIEW TEXT DATASET.

	PSNR	SSIM
Rainy	20.8128	0.7794
CNN [4]	17.6142	0.6196
DDN [5]	23.0897	0.8678
JORDER [16]	18.5631	0.7522
DID-MDN [17]	23.2442	0.8343
Ours	**24.7770**	**0.8705**

Acknowledgments

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan V GPU used for this research.

References

[1] The street view text dataset. `http://www.iapr-tc11.org/mediawiki/index.php?title=The_Street_View_Text_Dataset`.

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2010.

[3] David Eigen, Dilip Krishnan, and Rob Fergus. Restoring an image taken through a window covered with dirt or rain. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 633–640, 2013.

[4] Xueyang Fu, Jiabin Huang, Xinghao Ding, Yinghao Liao, and John Paisley. Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing*, 26(6):2944–2956, 2017.

[5] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3855–3863, 2017.

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[7] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

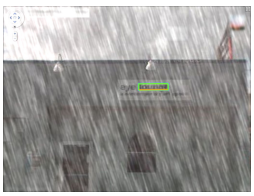
[9] Siyuan Li, Iago Breno Araujo, Wenqi Ren, Zhangyang Wang, Eric K Tokuda, Roberto Hirata Junior, Roberto Cesar-Junior, Ji-

-

bb1: THAL CUISINE

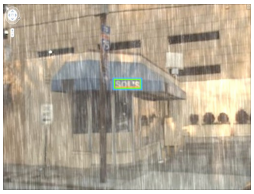
-

bb1: A
bb2: Budget



bb1: Tourige

bb1: >
bb2: eye lounge

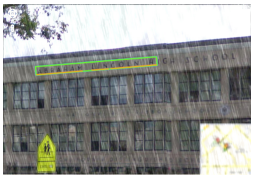


bb1: SOUS

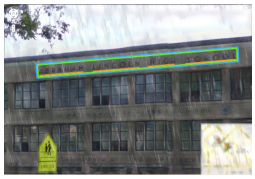
bb1: SOL'S

bb1: 1
bb2: +
bb3: z
bb4: RMISSION

bb1: < e >
bb2: VEEELER MISSION
MINISTRIES



bb1: ABRAHAM
LUN COLNA



bb1: ABRAHAM LINCOLN
HIGH SCHOOL

Figure 6: Some results over the Rainy Street View Text Dataset with the relative bounding boxes and detected texts.

awan Zhang, Xiaojie Guo, and Xiaochun Cao. Single image deraining: A comprehensive benchmark analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3838–3847, 2019.

[10] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.

[11] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[12] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3883–3891, 2017.

[13] Rui Qian, Robby T Tan, Wenhan Yang, Jiajun Su, and Jiaying Liu. Attentive generative adversarial network for raindrop removal from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2482–2491, 2018.

[14] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[15] Gerald Schaefer and Michal Stich. Ucid: An uncompressed color image database. In *Storage and Retrieval Methods and Applications for Multimedia 2004*, volume 5307, pages 472–480. International Society for Optics and Photonics, 2003.

[16] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1357–1366, 2017.

[17] He Zhang and Vishal M Patel. Density-aware single image deraining using a multi-stream dense network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 695–704, 2018.

[18] He Zhang, Vishwanath Sindagi, and Vishal M Patel. Image deraining using a conditional generative adversarial network. *IEEE transactions on circuits and systems for video technology*, 2019.

Author Biography

Simone Zini received the B.Sc. degree and the M.Sc. degree in computer science from the Department of Informatics, Systems and Communication (DISCo), University of Milano-Bicocca, Italy, respectively in 2015 and 2018. He is currently a Ph.D. student at DISCo, University of Milano-Bicocca. His current research interests concerns machine learning, image enhancement and computational photography.

Simone Bianco received the B.Sc. and M.Sc. degrees in mathematics from the University of Milano-Bicocca, Italy, in 2003 and 2006, respectively, and the Ph.D. degree in computer science from the Department of Informatics, Systems and Communication (DISCo), University of Milano-Bicocca, in 2010. He is currently an Assistant Professor. His current research interests include computer vision, machine learning, optimization algorithms, and color imaging.

Raimondo Schettini is a Full Professor with the University of Milano-Bicocca, Italy. He is the Head of the Imaging and Vision Laboratory. He has been associated with the Italian National Research Council, since 1987, where he has led the Color Imaging Laboratory, from 1990 to 2002. He has been the Team Leader on several research projects. He has authored or coauthored over 300 refereed papers and holds several patents on color reproduction, image processing, analysis, and classification. He is a fellow of the International Association of Pattern Recognition for his contributions to pattern recognition research and color image analysis.