Weather Attribute-Aware Multi-Scale Image Generation with Residual Learning

Wei-Ta Chu
National Cheng Kung University
Tainan, Taiwan
wtchu@gs.ncku.edu.tw

Li-Wei Huang
National Chung Cheng University
Chiayi, Taiwan
alta9160@gmail.com

***Abstract*—We present image generation networks to generate images conforming to specified weather attributes. Taking weather attributes as the conditions, the proposed networks generate scene images with the help of a guided reference image. To generate higher-resolution images, we construct a multi-scale generation framework consisting of a global generator and a local enhancer. Furthermore, we integrate the idea of residual learning into the proposed framework, and aim at generating fine-grained texture. The evaluation shows performance comparison both from quantitative and qualitative perspectives. A comprehensive study including the impact of different attributes and extension of the proposed models is also provided. This work is kind of hybrid approach among various image generation studies.**

I. Introduction

Image generation based on generative models has attracted much attention in recent years. As the rapid development of generative adversarial networks (GANs), many studies have been proposed to generate images. At the beginning, images are generated from random input vectors. Later, conditional GANs [10] were proposed to empower generative models with control functionality. With conditional GANs, user can change attributes to control generated images at his/her will. However, such generative model is still limited to low-resolution results.

Another popular image generation application is image style transfer [2]. Given a photo and a target painting, the style transfer model generates an image which visual content is similar to the photo but texture is similar to the painting. Guided by the painting, generation results are impressive.

In this paper, we propose an image generation scenario that is between conditional GANs and image style transfer. Given target attributes and a reference image, we build a generative model to generate images conforming to the attributes. This scenario is especially suitable to generate scene images where visual appearance changes as some physical attributes change. Fig. 1 illustrates the concept, where the generative model is guided by the reference image, and generates scene images that are in warm and wet spring or in winter with snow, according to given attributes. To enable high-resolution image generation, we take the idea of multi-scale generation, where a global generator generates rough scene layout, and a local enhancer improves results by considering information from the reference image. To further improve visual quality, the idea of residual learning is also integrated. The global generator focuses on predicting the residuals between synthesized images and real images, and the local enhancer is viewed to adjust the reference image based on predicted residuals.

978-1-7281-4999-8/20/$31.00 ©2020 IEEE

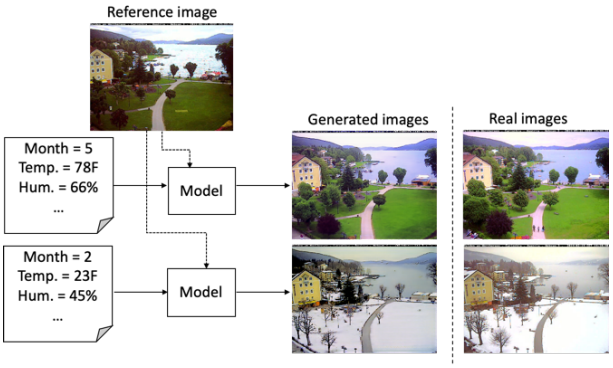


Fig. 1. Concept of the proposed image generation scenario. Starting from a reference image, the proposed model generates an image conforming to the desired attributes.

Contributions of this paper are threefold:

- We propose an image generation scenario where the inputs include a set of attributes and a reference image. This scenario would be practical in some applications. For example, when making a travel plan, the proposed model can be used to synthesize what the place will look like before visiting.
- We propose a guided multi-scale image generation framework with the idea of residual learning.
- We comprehensively evaluate the proposed methods from both qualitative and quantitative perspectives.

II. Related Works

Generative models have been studied for decades. Recently, generative adversarial networks (GANs) [3] pushed the studies to a new era. Given a noise vector $\boldsymbol{z}$, the distribution of training data $\boldsymbol{x}$ is estimated by a generator G, and a discriminator D is built to discriminate whether $G(\boldsymbol{z})$ comes from the distribution over $\boldsymbol{x}$. The objective of G is to maximize the probability of D making a mistake, while the objective of D is to maximize the probability of making a correct decision. To make generation more controllable, conditional GANs [10]

were proposed to make the generator and the discriminator condition on an auxiliary vector $\boldsymbol{y}$. The generator is to generate image $G(\boldsymbol{z}|\boldsymbol{y})$ that not only looks real but also conforms to the auxiliary vector $\boldsymbol{y}$, and the discriminator $D(\boldsymbol{z}|\boldsymbol{y})$ is to make discrimination as good as possible.

In addition to class labels, the auxiliary vector $\boldsymbol{y}$ can be from an observed image. For example, an image-to-image translation method developed in [6] took an input edge map as $\boldsymbol{y}$ and synthesized a photo-realistic image. When training the model, image pairs in two different domains (e.g., edge map and photo-realistic image) should be provided, which are not always available. In [8], an unsupervised image-to-image translation was proposed by assuming that corresponding images from two different domains can be mapped to a common latent space. A pair of encoders (for feature mapping), generators (for image construction/generation), and discriminators (for evaluating realistic or not) are constructed. Results of previous works were often limited to low resolution. Wang et al. [12] proposed a multi-scale generator and discriminator architecture to enable high-solution image generation. They decomposed the generator into a global generator and a local enhancer. The global generator outputs a lower-resolution image and feature maps, which is integrated into the intermediate feature maps of the local enhancer to provide finer enhancement.

Another popular generative scenario is image style transfer. Given two images, one is the source image $\boldsymbol{X}$ and another is the target image $\boldsymbol{Y}$, the style transfer model aims at generating an image $\hat{\boldsymbol{X}}$ which visual content is similar to $\boldsymbol{X}$ but its style is similar to $\boldsymbol{Y}$. The field of image style transfer has rapidly flourished since the power of convolutional neural networks (CNNs) on texture synthesis was described in [2].

The input of conditional GANs is an auxiliary vector or an image, and the inputs of image style transfer are two images. In our proposed scenario, an attribute vector and a reference image are fed to the proposed model. We integrate the concepts of residual learning and multi-scale generator/discriminator to generate images conforming to user-specified attributes.

III. Attribute-Aware Image Generation

A. The Baseline System

Given an attribute vector $\boldsymbol{w}$, we want to build a model $\mathcal{F}$ to output an image $\hat{\boldsymbol{x}} = \mathcal{F}(\boldsymbol{w})$ that is similar to the true image $\boldsymbol{x}$ with attributes $\boldsymbol{w}$. For example, we may like to generate a scene image in the summer, with temperature $70°F$ and humidity 60%.

We first develop a baseline system motivated by conditional GAN, as shown in Fig. 2. The training data are images in the same scene but in a variety of weather conditions. This conditional GAN consists of a generator G_1, a discriminator D_1, and a classifier C_1. Given an attribute vector $\boldsymbol{w}$, the objective of the generator G_1 is to generate realistic images $G_1(\boldsymbol{w})$, while the discriminator D_1 aims to distinguish real images from generated ones. A classifier C_1 is developed to predict attributes from $G_1(\boldsymbol{w})$ and $\boldsymbol{x}$, both of which should be close to $\boldsymbol{w}$. Given training data $\{\boldsymbol{x}, \boldsymbol{w}\}$, the baseline system is constructed by the following minimax game:

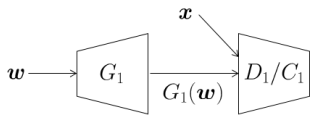


Fig. 2. Illustration of the baseline system.

$$\min_{G_1} \max_{D_1} \mathcal{L}_{adv}^{(1)} + \mathcal{L}_{con}^{(1)} + \mathcal{L}_{att}^{(1)}, \tag{1}$$

where the adversarial loss $\mathcal{L}_{adv}^{(1)}$ is given as

$$\mathcal{L}_{adv}^{(1)}(G_1, D_1) = \mathbb{E}_{\boldsymbol{x}}[\log D_1(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{w}}[\log(1 - D_1(G_1(\boldsymbol{w})))]. \tag{2}$$

The content loss $\mathcal{L}_{con}^{(1)}$ is defined as the L1 distance between $G_1(\boldsymbol{w})$ and $\boldsymbol{x}$. The attribute loss $\mathcal{L}_{att}^{(1)}$ is calculated as a combination of two L1 distances:

$$\mathcal{L}_{att}^{(1)}(G_1, C_1) = \frac{1}{2}\|C_1(G_1(\boldsymbol{w})) - \boldsymbol{w}\| + \frac{1}{2}\|C_1(\boldsymbol{x}) - \boldsymbol{w}\|, \tag{3}$$

where $C_1(G_1(\boldsymbol{w}))$ and $C_1(\boldsymbol{x})$ are the attributes predicted from the generated image and the real image $\boldsymbol{x}$, respectively. The input of C_1 is the feature maps of the second last layer of D_1, i.e., D_1 conceptually acts as a feature extractor for C_1. This input is then processed by a fully-connected layer to output attributes.

B. Guided Multi-Scale Generation (GMS)

Resolution of the images generated by the basic conditional GAN is limited. Motivated by [12], we develop a multi-scale generation framework to facilitate high-resolution generation. In addition, if we focus on a specific scene, the basic layout remain similar no matter how the weather conditions change. We thus propose to utilize a reference image to guide the generation process.

Taking a semantic label map as the input, the network proposed in [12] achieves image generation in a coarse-to-fine manner. The first sub-network acts as a global generator to roughly generate scene layout, while the second sub-network acts as a local enhancer to synthesize more details. Particularly, the global generator is constituted by a convolutional front-end $G_2^{(F)}$ followed by a set of residual blocks $G_2^{(R)}$, and then followed by a transposed convolutional back-end $G_2^{(B)}$ to generate a low-resolution result. The local enhancer take the semantic map of the original resolution as input, and is also constituted by a convolutional front-end $G_3^{(F)}$, a set of residual blocks $G_3^{(R)}$, and a transposed convolutional back-end $G_3^{(B)}$. The input to $G_3^{(R)}$ is the element-wise sum of the output feature map of $G_3^{(F)}$ and the last feature map of $G_2^{(B)}$. The local enhancer jointly considers global information and local details in the generation process.

Motivated by this idea, we propose the network structure shown in Fig. 3. Given an attribute vector $\boldsymbol{w}$, we utilize the baseline generator G_1 mentioned in Sec. III-A to generate a

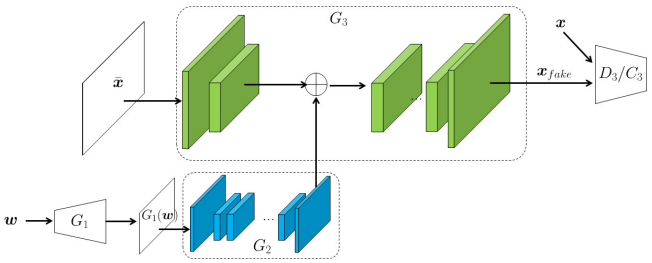


Fig. 3. Illustration of the guided multi-scale generation (GMS) model.

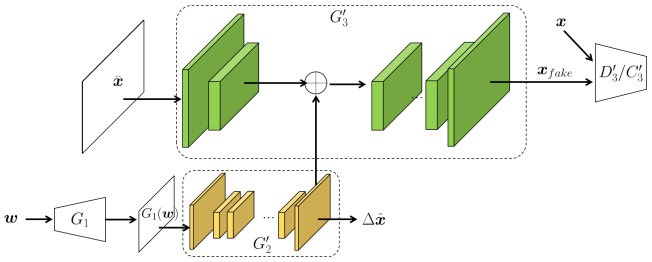


Fig. 4. Illustration of the guided multi-scale generation with residual learning (GMSR).

low-resolution image $G_1(\boldsymbol{w})$, which role is similar to the semantic map mentioned above and is fed to the global generator G_2. The last output feature maps of G_2 are integrated with the feature maps of $G_3^{(F)}$. To guide the local enhancer with texture details, we randomly select one image from the training data as the reference image $\bar{\boldsymbol{x}}$. The local enhancer G_3 extracts features from $\bar{\boldsymbol{x}}$. By integrating features from $\bar{\boldsymbol{x}}$ and $G_1(\boldsymbol{w})$, the local enhancer G_3 is built to generate high-resolution image.

The above model is constructed to make the generated image $\boldsymbol{x}_{fake} = G_3(\bar{\boldsymbol{x}}, G_2(G_1(\boldsymbol{w})))$ as realistic as possible. The discriminator D_3 is constructed to distinguish real images from generated ones. Similar to that in Fig. 2, a classifier C_3 is developed to predict attributes from $\boldsymbol{x}_{fake}$ and the real image $\boldsymbol{x}$, both of which should be close to $\boldsymbol{w}$. Given training data $\{\boldsymbol{x}, \boldsymbol{w}\}$, the guided multi-scale generation system is constructed by the following minimax game:

$$\min_{G_2,G_3} \max_{D_3} \mathcal{L}_{adv}^{(2)} + \mathcal{L}_{con}^{(2)} + \mathcal{L}_{att}^{(2)}, \tag{4}$$

where the adversarial loss $\mathcal{L}_{adv}^{(2)}$ is given as

$$\mathcal{L}_{adv}^{(2)}(G_2, G_3, D_3) = \mathbb{E}_{\boldsymbol{x}}[\log D_3(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{w}}[\log(1 - D_3(\boldsymbol{x}_{fake}))], \tag{5}$$

and the content loss $\mathcal{L}_{con}^{(2)}$ and the attribute loss $\mathcal{L}_{att}^{(2)}$ are defined as the same way in Sec. III-A.

C. Guided Multi-Scale Generation with Residual Learning (GMSR)

Motivated by the success of residual learning [4], we would like to investigate how if we make G_2 focus on predicting *the residual between the real image and the targeted generated image*, and let G_3 focus on adjusting the reference image into the final result. We slightly modify the framework shown in Fig. 3 and illustrate guided multi-scale generation with residual learning in Fig. 4.

The major difference comes from G_2. The G_2 in Fig. 3 is used to extract feature maps from the given low-resolution generated image. On the other hand, the global residual predictor G'_2 in Fig. 4 not only extracts feature maps from the input, but also predict the residual $\Delta\hat{\boldsymbol{x}}$ between the real image $\boldsymbol{x}$ and the generated image $\boldsymbol{x}_{fake}$. Given training data $\{\boldsymbol{x}, \boldsymbol{w}\}$, the GMSR model is constructed by finding the best settings via the following minimax game:

$$\min_{G'_2,G'_3} \max_{D'_3} \mathcal{L}_{adv}^{(3)} + \mathcal{L}_{con}^{(3)} + \mathcal{L}_{att}^{(3)} + \mathcal{L}_{\Delta}^{(3)}, \tag{6}$$

where the adversarial loss $\mathcal{L}_{adv}^{(3)}$ is given as

$$\mathcal{L}_{adv}^{(3)}(G'_2, G'_3, D'_3) = \mathbb{E}_{\boldsymbol{x}}[\log D'_3(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{w}}[\log(1 - D'_3(\boldsymbol{x}_{fake}))], \tag{7}$$

and the content loss $\mathcal{L}_{con}^{(3)}$ and the attribute loss $\mathcal{L}_{att}^{(3)}$ are defined as the same way in Sec. III-A. The residual loss $\mathcal{L}_{\Delta}^{(3)}$ is defined as

$$\mathcal{L}_{\Delta}^{(3)}(G'_2) = \|\Delta\boldsymbol{x} - \Delta\hat{\boldsymbol{x}}\|_2, \tag{8}$$

where $\Delta\boldsymbol{x}$ is the real residual calculated as the L2 distance $\|\boldsymbol{x} - \boldsymbol{x}_{fake}\|_2$, and $\Delta\hat{\boldsymbol{x}}$ is the predicted residual, i.e., $\Delta\hat{\boldsymbol{x}} = G'_2(G_1(\boldsymbol{w}))$.

D. Implementation Details

1) Evaluation Data: We collect evaluation data from the AMOS dataset [7]. The AMOS dataset consists of a large amount of images captured by 538 outdoor webcams in the United States. We select images of three scenes captured by camera #4829, #5454, and #8438. These scenes (Scene 1, Scene 2, and Scene 3) are picked because they have large visual variations in different seasons, and the cameras capture a wide range of scene rather than focusing on specific small objects. Fig. 5 shows sample images from these three scenes. Table I shows detailed information of these scenes. The images captured at 9am, 10am, ..., 15am on each day are taken to form the evaluation dataset.

2) Weather Attributes: According to each image's geographical information and timestamp, we can obtain weather attributes at a specific time instant from the closest meteorological station, based on the cli-MATE online service [1]. Table II shows the weather attributes and their numerical ranges. Because different attributes' numerical ranges and order of magnitude are different, we encode each attribute into a one-hot vector. For example, we encode a temperature value into a 12-dimensional one-hot vector. The quantization step is $10°F$. Temperature values $\leq -7°F$ are encoded as $\boldsymbol{w}_t = (100...000)$, temperature values from $-6°F$ to $3°F$ are encoded as $\boldsymbol{w}_t = (0100...000)$, values from $4°F$ to $13°F$ are encoded as $\boldsymbol{w}_t = (0010...000)$, and so on. For the month information, January is encoded as $\boldsymbol{w}_m = (100...000)$, February is encoded as $\boldsymbol{w}_m = (010...000)$, and so on. Finally, to represent an image's weather attribute, we concatenate

[1] http://mrcc.isws.illinois.edu/CLIMATE/

TABLE I
DETAILED INFORMATION OF THE EVALUATION DATA.

	Training Duration	# Training Images	Testing Duration	# Testing Images
Scene 1	2011/10/01 – 2014/02/28	5412	2014/03/01 – 2015/01/13	617
Scene 2	2012/06/25 – 2013/10/15	7012	2014/01/09 – 2014/12/03	599
Scene 3	2012/09/16 – 2013/08/20	5412	2011/08/15 – 2012/09/15	734

Fig. 5. Five sample images of three scenes, respectively. Scene 1, Scene 2, and Scene 3 are shown from top to down.

TABLE II
WEATHER ATTRIBUTES AND THEIR NUMERICAL RANGES AND DIMENSIONALITY.

Attributes	Numerical Range	Dimensionality
Month	1 to 12	12
Temperature ($^\circ F$)	< -7 to > 97	12
Relative humidity (%)	< 19 to > 91	10
Dew Point ($^\circ F$)	< -10 to > 71	11
Wind speed (mph)	< 8 to > 56	8
Wind direction (degree)	< 50 to > 300	7
Visibility (mile)	0 to > 9	11
		71

seven one-hot vectors to form a 71-dimensional vector $\boldsymbol{w} = (\boldsymbol{w}_m, \boldsymbol{w}_t, \boldsymbol{w}_h, ..., \boldsymbol{w}_v)$.

3) Model Training: Resolution of training images is 320×240. To construct the baseline model, we randomly initialize parameters of G_1, D_1, and C_1. The Adam optimizer is adopted to find good model parameters, and the learning rate is set as 0.0002. Similar to the setting mentioned in [13], for each training iteration we just take one sample to update model parameters, i.e., mini-batch size is 1. The model is trained in an end-to-end way, but for each mini-batch, parameters of the generator G_1 are updated three times, followed by updating parameters of D_1 and C_1 once. The baseline model is trained for 20 epochs.

In the GMS model, we take G_1 of the baseline model as a low-resolution image generator. Before training the GMS model, we first re-train the baseline model based on down-sampled 160×120 training images. The obtained G_1 is then used in the GMS model. Similarly, we randomly initialize G_2, G_3, D_3, and C_3, and update model parameters by the Adam optimizer. Model parameters of the adopted G_1 are fixed, and for each mini-batch parameters of the G_2 and G_3 are updated three times, followed by updating parameters of D_3 and C_3 once. The GMS model is also trained for 20 epochs. The training settings for the GMSR model are quite the same.

IV. EVALUATION

A. Performance Comparison

1) Evaluation based on Distribution Generation: We evaluation performance based on Inception Score (IS) [11] and Frechet Inception Distance (FID)[5]. IS was designed to jointly consider quality and diversity of generation results. Better models are assumed to generate results, when evaluated by the classifier, with low-entropy class distribution. In addition, better models are assumed to generate results evenly covering all classes. A larger IS value means better generation results. The assumptions for calculating IS values are not always true. Therefore, FID was designed to consider the difference in embedding of true and fake data [9]. A smaller FID value means better generation results.

Average IS and FID values of all testing data for each scene are calculated to show overall performance. Table III shows performance comparison between [13] and the three models mentioned above. In terms of IS values, the GMS model outperforms the baseline model in two of three scenes, which shows the positive effect of guided multi-scale generation. The GMSR model with residual learning consistently improves performance. In terms of FID values, the GMSR model largely surpasses the GMS model and the baseline model.

The work [13] was proposed to generate high-resolution texture based on a small patch, with joint consideration of visual content loss and style loss. It really achieves higher IS values in three scenes. However, the proposed GMSR model works significantly better than [13] in terms of FID values.

2) Evaluation based on Attribute Prediction: As we aim at attribute-aware image generation, better attribute recognition should be obtained if images are generated well. Here we adopt the state-of-the-art visual weather predictor mentioned in [1] to predict temperature and humidity. The L1 distance between temperature (humidity) predicted from $\boldsymbol{x}_{fake}$ and that from $\boldsymbol{x}$ is calculated. Notice that we do not compare the

TABLE III
PERFORMANCE COMPARISON BETWEEN DIFFERENT MODELS, IN TERMS OF IS AND FID.

	IS			FID		
	Scene 1	Scene 2	Scene 3	Scene 1	Scene 2	Scene 3
[13]	**2.31**	**1.57**	**1.66**	236.18	205.07	160.45
[13] w. attribute loss	1.97	1.54	1.63	220.80	200.89	130.94
Baseline	1.23	1.43	1.44	262.32	148.28	127.92
GMS	1.57	1.33	1.62	191.07	77.1	134.21
GMSR	**1.60**	**1.50**	**1.66**	**106.8**	**68.71**	**88.01**

TABLE IV
PERFORMANCE COMPARISON BETWEEN DIFFERENT MODELS, IN TERMS OF TEMPERATURE AND HUMIDITY ESTIMATION ERRORS.

	Temperature Error			Humidity Error		
	Scene 1	Scene 2	Scene 3	Scene 1	Scene 2	Scene 3
Baseline	3.26	2.23	3.21	1.87	1.33	1.33
GMS	3.32	1.78	2.75	2.39	1.30	1.74
GMSR	**3.26**	**1.54**	**2.38**	**1.86**	**1.25**	**1.06**

temperature (humidity) predicted from $\boldsymbol{x}_{fake}$ with the true temperature (humidity) associated with $\boldsymbol{x}$, because currently visual weather prediction is still not a highly-accurate task. According to [1], the average prediction error of the state-of-the-art model is still over $4°$C, and thus this model is not stable enough to be the oracle. Before a better predictor is proposed, we adopt this approach as a compromise.

Table IV shows that, in terms of temperature error, the GMS model works better than the baseline in two of three scenes; while in terms of humidity error, the GMS model works better than the baseline in only one of three scenes. Although we often can perceive finer texture details from the images generated by the GMS model (see Fig. 6), the prediction performance may not be consistent with human perception. On the other hand, from Table IV we see that the GMSR model consistently works the best in three scenes.

3) Qualitative Analysis: Fig. 6 shows sample images generated by three different models. For each row, the first subfigure is the given weather attributes, the second one is the ground truth image associated with the given attributes, and the remaining images are generated by the baseline, the GMS, and the GMSR models, respectively. For the first row, we see that the image generated by the baseline is blurry, and the one generated by the GMS model is much clearer. The GMSR model generates the finest texture. For the second row, the difference between three models is not as apparent as that in the first row. But we still can see that fine-grained texture can be generated by the GMSR model if we particularly enlarge parts of the generated images. The situation in the third row is similar to that in the first row.

B. Discussion

We would like to investigate that, among the seven weather attributes, which ones are more influential to image generation? We purposely change a specific weather attribute to see how the generation results change. Given a set of weather attributes, say (month=8, temperature=67°F, humidity=90%, dew point=64°F, wind speed=5 mph, wind direction = 120 degree, and visibility = 10 miles), we purposely change month to January (month=1), March (month=3), ..., and November, and generate results shown in the first row of Fig. 7, respectively. Other attributes are fixed in generating these six results. Interestingly, when month changes from the winter to the summer, the generated scene image changes as we expect. We see accumulated snow in January, melting snow in March, green grass and flourishing trees in May, July, and September, and trees with withered leaves in November.

Given weather attributes, say (month=3, temperature=61°F, humidity=22%, dew point=22°F, wind speed=14 mph, wind direction = 180 degree, and visibility = 10 miles), we purposely change temperature to 30°F, 40°F, 50°F, and 60°F, and generate results shown in the second row of Fig. 7. We clearly see that the snow melts as temperature increases.

Given attributes (month=3, temperature=53°F, humidity=76%, dew point=46°F, wind speed=6 mph, wind direction = 90 degree, and visibility = 7 miles), we purposely change wind direction to 50 degree, 90 degree, ..., and 250 degree, and generate results shown in the third row of Fig. 7. Although the area of melting snow changes a little bit, basically different results are quite similar.

Quantitatively, we calculate the root mean square error (RMSE) between every possible pair (based on pixel intensity) in each row of Fig. 7, and then calculate the average RMSEs. Finally, the average RMSEs of these three rows are 59.47, 38.87, and 25.35, respectively. The largest average RMSE for the first row shows that the diversity of generated images according to time change is the largest. Both from the qualitative and quantitative perspective, we see that time and temperature information is more influential to image generation.

V. CONCLUSION

We have presented an attribute-aware image generation network that jointly considers multi-scale generators and residual learning. Given a set of attributes, a generative adversarial network guided by the given reference image can generate images conforming to the targeted attributes. To generate high-resolution results, a multi-scale generation framework consisting of a global generator and a local enhancer is constructed. Furthermore, with the idea of residual learning, we transform the global generator to generating residuals, which

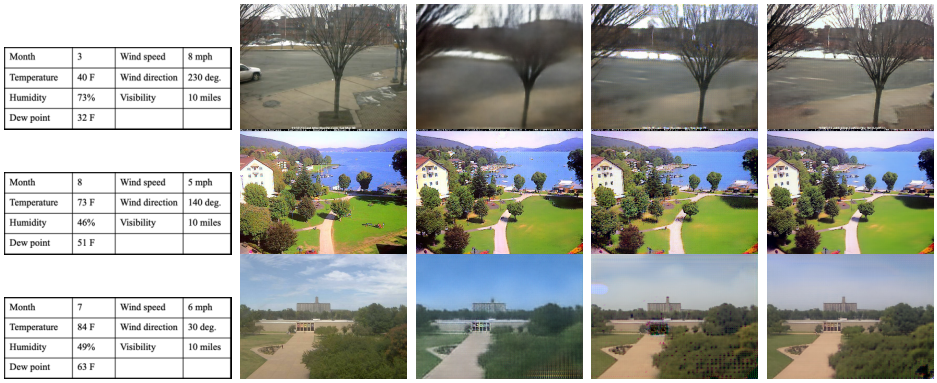
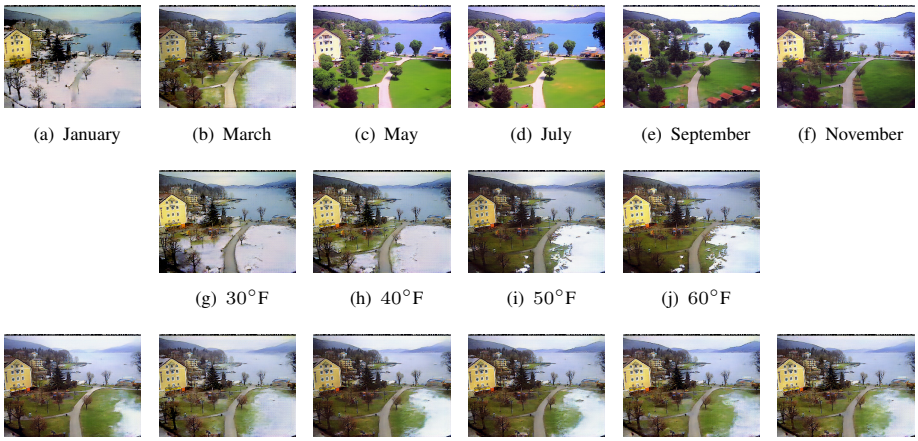


Fig. 6. Sample generation results. At each row, from left to right: weather attributes, ground truth, the image generated by the baseline model, the image generated by the GMS model, and the image generated by the GMSR model.



(a) January (b) March (c) May (d) July (e) September (f) November

(g) 30°F (h) 40°F (i) 50°F (j) 60°F

(k) 50 deg. (l) 90 deg. (m) 130 deg. (n) 170 deg. (o) 210 deg. (p) 250 deg.

Fig. 7. Generation results by changing a specific weather attribute.

is demonstrated to be more effective to generate high-quality images. Comprehensive evaluation from both qualitative and quantitative perspectives are provided.

Acknowledgment

This work was partially supported by the Ministry of Science and Technology, Taiwan, under the grant 108-2221-E-006-227-MY3, 107-2221-E-006-239-MY2, 107-2923-E-194-003-MY3, 107-2627-H-155-001, and 107-2218-E-002-055.

References

[1] Chu, W.T., Ho, K.C., Borji, A.: Visual weather temperature prediction. In: Proceedings of IEEE Winter Conference on Applications of Computer Vision. pp. 234–241 (2018)
[2] Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (2016)
[3] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proceedings of Advances in Neural Information Processing Systems (2014)
[4] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (2016)
[5] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Proceedings of Neural Information Processing Systems (2017)
[6] Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial nets. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (2017)
[7] Jacobs, N., Roman, N., Pless, R.: Consistent temporal variations in many outdoor scenes. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (2007)
[8] Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Proceedings of Advances in Neural Information Processing Systems (2017)
[9] Lucic, M., Kurach, K., Michalski, M., Bousquet, O., Gelly, S.: Are gans created equal? a large-scale study. In: Proceedings of Neural Information Processing Systems (2018)
[10] Mirza, M., Osindero, S.: Conditional generative adversarial nets (2014), https://arxiv.org/abs/1411.1784
[11] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Proceedings of Neural Information Processing Systems (2016)
[12] Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (2018)
[13] Zhou, Y., Zhu, Z., Bai, X., Lischinski, D., Cohen-Or, D., Huang, H.: Non-stationary texture synthesis by adversarial expansion. ACM Transactions on Graphics **37**(4) (2018)