PI-TRANS: PARALLEL-CONVMLP AND IMPLICIT-TRANSFORMATION BASED GAN FOR CROSS-VIEW IMAGE TRANSLATION

Bin Ren[1,2], *Hao Tang*[3], *Yiming Wang*[4], *Xia Li*[3], *Wei Wang*[5], *Nicu Sebe*[2]

[1]University of Pisa, [2]University of Trento, [3]ETH Zurich, [4]FBK, [5]Beijing Jiaotong University

ABSTRACT

For semantic-guided cross-view image translation, it is crucial to learn where to sample pixels from the source view image and where to reallocate them guided by the target view semantic map, especially when there is little overlap or drastic view difference between the source and target images. Hence, one not only needs to encode the long-range dependencies among pixels in both the source view image and target view semantic map but also needs to translate these learned dependencies. To this end, we propose a novel generative adversarial network, PI-Trans, which mainly consists of a novel Parallel-ConvMLP module and an Implicit Transformation module at multiple semantic levels. Extensive experimental results show that PI-Trans achieves the best qualitative and quantitative performance by a large margin compared to the state-of-the-art methods on two challenging datasets. The source code is available at https://github.com/Amazingren/PI-Trans.

Index Terms— Cross-view Image Translation, GANs, MLP

1. INTRODUCTION

Semantic-guided cross-view image translation aims at generating images from a source view to a different target view given a target view semantic map as the guidance. In particular, we focus on the cases of translating from the aerial-view to the ground-view for photo-realistic urban scene synthesis, which can be beneficial for geo-localization [1–7] or civil engineering design with the semantic map either being extracted from another modality or being designed [8–11].

However, translating images from two distinct views with little overlap is a challenging problem, as the area coverage, the appearances of objects, and their geometrical arrangement in the ground-view image can be extremely different from the aerial-view image (see the comparison between the 1st column and the 3rd column in Fig. 1). Early works usually adopt the convolutional neural networks (CNN) based encoder-decoder structure [9, 12] with the generative adversarial networks (GANs) [13, 14]. However, such methodology
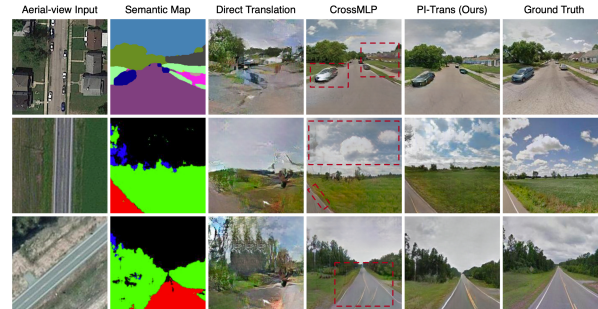
Fig. 1: Examples of the cross-view translation with different methods. The aerial-view image (column 1) and semantic map (column 2) serve as the inputs. Direct translation (column 3) generates the target ground view without the guidance of the semantic map. CrossMLP [15] (column 4) is the most recent state-of-the-art method, yet its results still contain some obvious artifacts marked with dotted red boxes. Our PI-Trans (column 5) generates results that are most similar to the ground-truth images (column 6).

suffers from satisfactory results when there exists little overlap between two views since the CNN-based methods struggle to establish the long-range relation due to the design nature of the convolutional kernels.

Usually, it's not easy to generate a photo-realistic image when just following the direct translation setting (the direct translation means that the generated ground-view image I'_g is directly generated from the aerial view image I_a without the assistance of the semantic map S_g, see the direct translation branch in Fig. 2). To improve the quality of the cross-view translation task, previous methods [10, 15, 16] took the target-view semantic map into consideration.

Though very insightful explorations had been performed, we find there are still some limitations that hinder the improvement of the quality of the generated images: (i) The pure CNN based methods (*i.e.*, [10, 16]) are difficult to establish the long-range relation due to its natural physical design of the convolutional kernels. (ii) The heavy fully-connected layers relied method [15] is subject to be insufficient for modeling the fine spatial information. (iii) All these three state-of-the-art methods mentioned above missed utilizing the very crucial but the easiest to be ignored *direct translation information* (the direct translation branch is shown in Fig. 2).
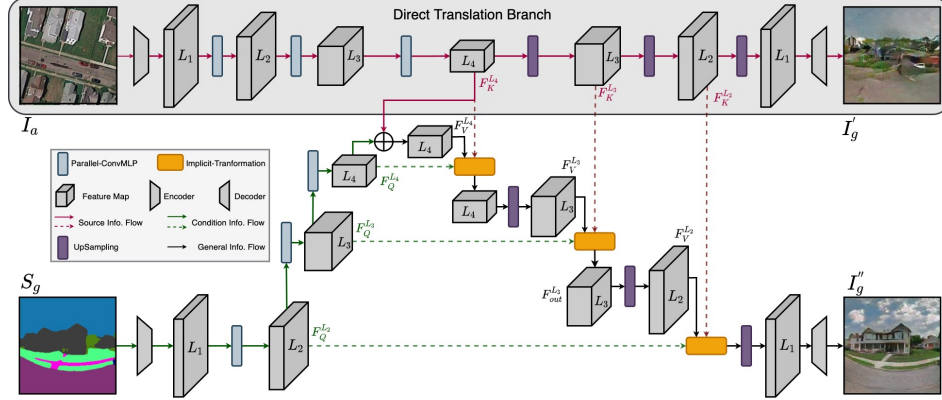
Fig. 2: Architecture of our PI-Trans. The proposed parallel-ConvMLP module depicted in blue is assembled in two encoder branches for effectively modeling both the long-range relation and the fine spatial information, the proposed implicit-transformation module that depicted in yellow is used in two decoder branches for transforming the source-view information to target-view at multiple feature levels. $\oplus$ denotes the element-wise addition. $I_g^{'}$ is the output of the direct translation branch, $I_g^{''}$ is the final output of the proposed PI-Trans.

To this end, we propose the Parallel-ConvMLP and Implicit-Transformation based GAN (PI-Trans), and its structure is shown in Fig. 2, which is mainly built with two encoder branches and two decoder branches. In the encoder branches, we propose the novel parallel-ConvMLP module that can effectively manage both the latent long-range relation and the detailed spatial information with its unique spatial-channel MLP in a novel parallel manner instead of the sequential order like [15, 17]. Besides, the input feature is uniformly split into two chunks, and each is fed to its corresponding MLP. By doing so, we implicitly introduced channel shuffling, thus being beneficial for long-range reasoning. In the decoder branches, unlike previous methods that just take the combined feature from the source-view image and the target-view semantic map for recovering the final target-view image, we, for the first time, add the direct translation information (for providing reasonable color distribution) via the proposed implicit-transformation modules at multiple feature levels.

To summarize, our contributions are listed as follows:

- We propose PI-Trans, which uses the transformation information that is directly learned from the source view to the target-view images to boost the performance.
- For modeling the long-range relation and the fine spatial information, we propose a novel parallel-ConvMLP module with an effective combination of CNN and MLP. Besides, an implicit-transformation module that conducts attention-based fusion at multiple feature levels is also proposed for a better translation result.
- Experimental results show that our method generates photo-realistic target-view ground images, scoring new state-of-the-art numerical results on two challenging datasets.

2. THE PROPOSED PI-TRANS

The architecture of our PI-Trans is shown in Fig. 2, which consists of two encoder branches and two decoder branches.

PI-Trans takes as input both the source-view aerial image $I_a \in \mathbb{R}^{3 \times H \times W}$ and the conditional target-view ground semantic map $S_g \in \mathbb{R}^{3 \times H \times W}$ at two encoder branches. First I_a and S_g are processed by two encoders to semantic L_1 level with dimension $(C_{L_1}, H/2, W/2)$, where C, H, and W mean the channel number, height, width, respectively. We then use the proposed parallel-ConvMLP module (Sec. 2.1) to further encode the L_1 level feature to L_2 ($2C_{L_1}$, $H/4$, $W/4$), L_3 ($4C_{L_1}$, $H/8$, $W/8$), and L_4 ($8C_{L_1}$, $H/16$, $W/16$) semantic levels. Unlike previous methods [15, 16] which generate the final target-view ground image $I_g^{''}$ only based on the combined feature coming from I_a and S_g at L_4 level (*i.e.*, $F_V^{L_4}$). We exploit another pathway, the direct translation branch, that directly produces a ground image $I_g^{'}$ at the target view from the source view, without interacting with the conditional semantic map. Then we use this direct transformation information accompanied with the semantic feature at the lower target pathway via our proposed implicit-transformation (Sec. 2.2) at 3 semantic levels (L_2, L_3, and L_4).

2.1. Parallel-ConvMLP Module

Given one pixel, it's extremely significant for the cross-view image translation task to understand which other pixels are related to it, or which object it belongs to. However, CNN kernels (Fig. 3(a)) are not good at modeling the long-range relation because of the locality of the fixed kernels. Therefore, the MLP-based methods MLP-Mixer [17] (Fig. 3(b)), ConvMLP [18] (Fig. 3(c)), and CrossMLP [15] (Fig. 3(d)) were proposed to ease this problem. However, both MLP-Mixer and CrossMLP are computationally heavy and are not subject to modeling the fine spatial patterns. ConvMLP [18] tackles this problem by combining a depth-wise convolution between two channel-wise MLPs. Yet, its performance is still unsatisfactory for the cross-view translation task (See Sec. 3).

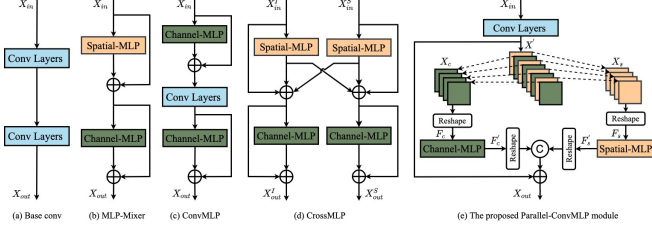To explore the balance between the long-range relation

Fig. 3: Illustrations of four comparison methods *i.e.*, (a) Basic convolutional layers, (b) MLP-Mixer [17], (c) ConvMLP [18], (d) CrossMLP module [15] and (e) our parallel-ConvMLP. The symbols $\oplus$ and $\copyright$ denote the element-wise addition and channel-wise concatenation.

and the spatial pattern modeling, we propose a parallel-ConvMLP module, which is shown in Fig. 3(e). Given an input feature X_{in} in dimension (c, h, w), it firstly goes through two convolutional layers:

$$X^{'} = \text{Conv}(\text{downConv}(X_{in})), \tag{1}$$

The first $(\text{downConv}(\cdot))$ is a strided convolutional layer while the second $(\text{Conv}(\cdot))$ is a normal one. $X^{'}$ is the output in dimension $(2c, h/2, w/2)$. This operation ensures the latent appearance and the local spatial pattern can be well established. Different from previous methods that simultaneously model the long-range relation and the spatial information in a sequential way [15, 17], we decouple the problem by dividing $X^{'}$ along channel dimension into two parts based on the parity of the channel index:

$$X_c = X^{'}_{2i-1}, \;\; X_s = X^{'}_{2i}, \;\; for \;\; i = 1, 2, 3, \cdots, 2c. \tag{2}$$

After that, we first flatten X_c and X_s from $(c, h/2, w/2)$ to (n, c) and (c, n), forming F_c and F_s ($n=w/2 \times h/2$). Then two fully-connected MLPs (channel-wise and spatial-wise) are applied to F_c and F_s. And each MLP block contains two fully-connected layers and a nonlinear GELU [19] activation function. These operations are formulated as follows:

$$\begin{aligned} F^{'}_c = \boldsymbol{W^c_2}\sigma(\boldsymbol{W^c_1}(F_c)_{*,i}), \;\; for \;\; i = 1, 2, 3, \cdots, c, \\ F^{'}_s = \boldsymbol{W^s_2}\sigma(\boldsymbol{W^s_1}(F_s)_{*,j}), \;\; for \;\; j = 1, 2, 3, \cdots, n, \end{aligned} \tag{3}$$

where $\boldsymbol{W^c_1}$ and $\boldsymbol{W^c_2}$ denote the learnable weights for the channel-wise MLP, while $\boldsymbol{W^s_1}$ and $\boldsymbol{W^s_2}$ are for the spatial-wise MLP. $\sigma(\cdot)$ means the GELU activation function. Then $F^{'}_c$ and $F^{'}_s$ are reshaped back to (c, h, w) and concatenated together. Finally, a skip-connection is used to add the concatenated feature to $X^{'}$:

$$X_{out} = X^{'} + \text{Cat}(\text{Reshape}(F^{'}_c), \text{Reshape}(F^{'}_s)), \tag{4}$$

where $\text{Cat}(\cdot)$ and $\text{Reshape}(\cdot)$ denote the concatenate and reshape operations.

2.2. Implicit Transformation Module

Since the performance for the direct translation ($I_a \rightarrow I^{'}_g$) is in a bad condition (see $I^{'}_g$ in Fig. 2 or the third column

in Fig. 1). Hence, previous methods SelectionGAN [16] and CrossMLP [15] ignored the direct translation branch and just used the combined feature $F^{L_4}_V$. Instead, we explore how this kind of latent information within the direct transformation branch may impact the overall generation performance since it provides a reasonable color distribution though bad in spatial structure. Therefor, we propose the implicit transformation module (the yellow blocks in Fig. 2), which conducts the fusion at multiple semantic levels.

There are three implicit transformation modules at L_4, L_3, and L_2 levels, each of them takes as input three kinds of information, *i.e.*, the ground semantic map feature F_Q, the directly translated feature F_K, and an extra input feature F_V. In Fig. 2, we visualize these three kinds of information flows in green, red, and black colors, respectively. The main idea behind the proposed implicit transformation module is to enable the transformed feature F_K to provide useful latent appearance or color information for generation. More specifically, we exploit the semantic map feature F_Q and the directly transformed feature F_K to construct an attention map with a softmax function. This operation uses the target-view semantic feature F_Q to select the most important information in the directly transformed feature F_K, which can also be seen as a learned latent transformation pattern. This attention map is then used to activate the most relevant feature in F_V, achieving the feature-level implicit transformation guided by the attention map. Finally, we exploit a skip-connection to maintain the result in the last module. And we take the L_3 level module for a detailed description. Given three inputs $F^{L_3}_Q$, $F^{L_3}_K$, and $F^{L_3}_V$, we first reshape them from (b, c, h, w) to $(b, c/4, n)$, $(b, c/4, n)$, and (b, c, n). Here $n=h \times w$. Then we adopt the residual attention mechanism to learn the fused feature:

$$F^{L_3}_{out} = F^{L_3}_V + \text{softmax}(F^{L_3}_Q(F^{L_3}_K)^T)F^{L_3}_V, \tag{5}$$

where $F^{L_3}_{out}$ denotes the output. It serves as the value feature for the implicit translation module at the next semantic level.

2.3. Discriminator and Optimization Objective

Discriminator. Following [10] and [16], the discriminator in the direct transformation branch takes the real image I_a and the generated image $I^{'}_g$ or the ground-truth image I_g as input. While for the lower decoder branch, it accepts the real image I_a and the generated image $I^{''}_g$ or the real image I_g as input.
Optimization Objective. The full optimization objective is:

$$\min_{\{G\}} \max_{\{D\}} \mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_{cGAN} \mathcal{L}_{cGAN} + \mathcal{L}_{tv} + \lambda_{per} \mathcal{L}_{per}, \tag{6}$$

where $\mathcal{L}_1$ denotes the pixel-level loss, $\mathcal{L}_{cGAN}$ denotes the adversarial loss, which is used for distinguishing the synthesized images pairs $(I_a, I^{''}_g)$ from the real image pairs (I_a, I_g). $\mathcal{L}_{tv}$ is the total variation regularization [20] and $\mathcal{L}_{per}$ is the perception loss which is commonly used for the generative

Table 1: Quantitative results on the Dayton dataset.

Method	Accuracy (%) ↑				Inception Score ↑			KL ↓	LPIPS ↓
	Top-1		Top-5		all	Top-1	Top-5		
Pix2pix [26]	6.80*	9.15*	23.55*	27.00*	2.8515*	1.9342*	2.9083*	38.26 ± 1.88*	-
X-Fork [10]	30.00*	48.68*	61.57*	78.84*	3.0720*	2.2402*	3.0932*	6.00 ± 1.28*	-
X-Seq [10]	30.16*	49.85*	62.59*	80.70	2.7384*	2.1304*	2.7674*	15.93 ± 1.32*	-
SelectionGAN [16]	42.11	68.12	77.74	92.89	3.0613	2.2707	3.1336	2.7406 ± 0.8613	0.7961
CrossMLP [15]	47.65	78.59	80.04	94.64	3.3466	2.2941	3.3783	2.3301 ± 0.8014	0.3750
PI-Trans (Ours)	**49.48**	**79.98**	**82.45**	**96.52**	**3.6713**	**2.4918**	**3.6824**	**2.0790 ± 0.6509**	**0.3656**
Real Data	-	-	-	-	3.8326	2.5781	3.9163	-	-

Table 2: Quantitative results on the CVUSA dataset.

Method	Accuracy (%) ↑				Inception Score ↑			KL ↓	LPIPS ↓
	Top-1		Top-5		all	Top-1	Top-5		
Pix2pix [26]	7.33*	9.25*	25.81*	32.67*	3.2771*	2.2219*	3.4312*	59.81 ± 2.12*	-
X-Fork [10]	20.58*	31.24*	50.51*	63.66*	3.4432*	2.5447*	3.5567*	11.71 ± 1.55*	-
X-Seq [10]	15.98*	24.14*	42.91*	54.41	3.8151*	2.6738*	4.0077*	15.52 ± 1.73*	-
SelectionGAN [16]	41.52	65.61	74.32	89.66	3.8074	2.7181	3.9197	2.9603 ± 0.9714	0.4122
CrossMLP [15]	44.96	69.96	76.98	91.91	3.8392	2.8035	3.9757	2.6903 ± 0.9432	0.3974
PI-Trans (Ours)	**47.87**	**74.57**	**80.36**	**94.68**	**4.1701**	**2.9878**	**4.2071**	**2.2363 ± 0.7759**	**0.3784**
Real Data	-	-	-	-	4.8749	3.3053	4.9938	-	-

Fig. 4: Qualitative results of different methods on Dayton.

Fig. 5: Qualitative results of different methods on CVUSA.

Table 3: Ablation results on Dayton-Ablation.

Method	Accuracy (%) ↑				Inception Score ↑			KL ↓	LPIPS ↓
	Top-1		Top-5		all	Top-1	Top-5		
A: Basic Conv	43.61	71.85	76.56	91.55	2.8132	2.1399	2.8036	2.9343 ± 0.9248	0.4392
B: MLP-Mixer	44.36	72.08	77.57	92.44	3.3207	2.2852	3.3282	2.5810 ± 0.8489	0.3926
C: ConvMLP	42.75	69.35	76.49	89.82	3.2442	2.2229	3.2580	2.8470 ± 0.8619	0.4093
D: CrossMLP	43.41	70.73	76.84	91.77	3.3241	2.2580	3.3340	2.7250 ± 0.8703	0.3987
E: P-ConvMLP	45.63	73.04	80.02	92.62	3.2573	2.2742	3.2957	2.4877 ± 0.8014	0.3917
F: PI-Trans	**49.90**	**79.27**	**82.83**	**95.20**	**3.3975**	**2.3461**	**3.4111**	**2.0856 ± 0.7045**	**0.3828**
Real Data	-	-	-	-	3.8307	2.5749	3.9159	-	-

tasks [20, 21] to make the produced images look more natural and smooth. We set λ_1, λ_{cGAN}, and λ_{per} to 100, 5, and 50, respectively.
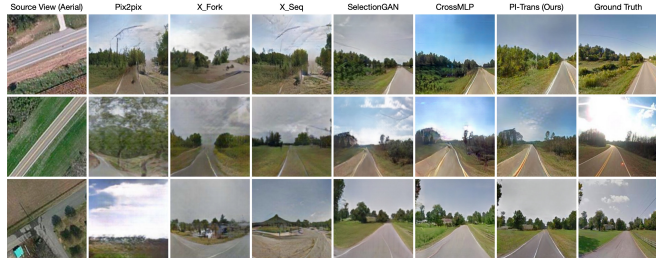
3. EXPERIMENTS

Datasets and Evaluation Metrics. We conduct experiments on two challenging public datasets, *i.e.*, Dayton [22], and CVUSA [23], following the similar settings as in [10,16]. We follow a similar evaluation protocol as in [10,16] to measure the quality of the generated ground images.

Implementation Details. *Encoders* are the same for both I_a and S_g, and each consists of one two-strided down-sampling convolutional layer (followed by batch normalization and ReLU activation function) and two one-strided ones (filter numbers: 16, 32, 32) to generate the L_1 level features. Given an input in dimension $(b, 3, 256, 256)$, the output is in dimension $(b, 32, 128, 128)$, where b denotes the batch size. Each *Up-sampling block* (depicted in purple in Fig. 2 are used in the decoder branches) consists of one nearest up-sampling layer, two one-strided convolutional layers (kernel size: 3, 3). *Decoder* consists of three one-strided convolutional layers (kernel size: 3, 3, and padding: 1, 1) and 1 Tanh activation function.

Training Settings. Following [10] and [16], Adam [24] is used as the solver with momentum terms $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The initial learning rate for is set to 0.0002. In addition, we train the proposed PI-Trans 35 epochs on the Dayton dataset and 30 epochs on the CVUSA dataset in an end-to-end manner. The code is implemented in PyTorch [25].

Experimental Results. PI-Trans is compared with five state-of-the-art methods, including Pix2pix [26], X-Fork [10], X-Seq [10], SelectionGAN [16], and CrossMLP [15]. Table 1 and Table 2 show the numerical results on Dayton and CVUSA. PI-Trans achieves the best performance on all the metrics with a significant improvement. The qualitative results are shown in Fig. 4 and Fig. 5 for Dayton and CVUSA, respectively. The target-view ground images generated by our PI-Trans are more natural and sharper.

Ablation Study. To validate the effectiveness of each proposed module, we select 1/3 samples randomly from the entire Dayton dataset, forming the Dayton-Ablation dataset. Six baselines *i.e.*, A (Basic Conv), B (MLP-Mixer), C (ConvMLP), D (CrossMLP), E (Parallel-ConvMLP), and F (PI-Trans) are proposed. For baseline A, we only use the convolutional layers, and without the implicit transformation modules. The output directly comes out from a combination of the features between source-view image features and the semantic map features. Baseline B, C, D, and E follow a similar setting as baseline A, with the only difference in the encoder branches. F is our full model. The results shown in Table 3 demonstrate that both our Parallel-ConvMLP (E) and implicit transformation (F) can boosts the quality of the generated ground image by a large margin.

4. CONCLUSION

We propose a new PI-Trans for generating realistic cross-view images. Significantly, the parallel-ConvMLP module is designed for balancing the relationship between latent long-range dependency modeling and spatial information maintenance. The implicit transformation module we designed at multiple semantic levels in this paper carefully takes care of not only the target-view semantic map feature and the source-view aerial image feature but also the direct translation information that is usually ignored by previous methods. We validate the effectiveness and advances of our proposed modules over existing methods.

5. REFERENCES

[1] Yujiao Shi, Liu Liu, Xin Yu, and Hongdong Li, "Spatial-aware feature aggregation for image based cross-view geo-localization," *NeurIPS*, vol. 32, 2019.

[2] Yujiao Shi, Xin Yu, Liu Liu, Tong Zhang, and Hongdong Li, "Optimal feature transport for cross-view image geo-localization," in *AAAI*, 2020, vol. 34, pp. 11990–11997.

[3] Liu Liu, Hongdong Li, and Yuchao Dai, "Stochastic attraction-repulsion embedding for large scale image localization," in *ICCV*, 2019, pp. 2570–2579.

[4] Sijie Zhu, Taojiannan Yang, and Chen Chen, "Vigor: Cross-view image geo-localization beyond one-to-one retrieval," in *CVPR*, 2021, pp. 3640–3649.

[5] Liu Liu and Hongdong Li, "Lending orientation to neural networks for cross-view geo-localization," in *CVPR*, 2019, pp. 5624–5633.

[6] Sixing Hu, Mengdan Feng, Rang MH Nguyen, and Gim Hee Lee, "Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization," in *CVPR*, 2018, pp. 7258–7267.

[7] Sudong Cai, Yulan Guo, Salman Khan, Jiwei Hu, and Gongjian Wen, "Ground-to-aerial image geo-localization with a hard exemplar reweighting triplet loss," in *ICCV*, 2019, pp. 8391–8400.

[8] Aysim Toker, Qunjie Zhou, Maxim Maximov, and Laura Leal-Taixé, "Coming down to earth: Satellite-to-street view synthesis for geo-localization," in *CVPR*, 2021, pp. 6488–6497.

[9] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros, "View synthesis by appearance flow," in *ECCV*, 2016.

[10] Krishna Regmi and Ali Borji, "Cross-view image synthesis using conditional gans," in *CVPR*, 2018, pp. 3501–3510.

[11] Mehmet Ozgur Turkoglu, William Thong, Luuk Spreeuwers, and Berkay Kicanaoglu, "A layer-based sequential framework for scene generation with gans," in *AAAI*, 2019, vol. 33, pp. 8901–8908.

[12] Jimei Yang, Scott E Reed, Ming-Hsuan Yang, and Honglak Lee, "Weakly-supervised disentangling with recurrent transformations for 3d view synthesis," in *NeurIPS*, 2015.

[13] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C Berg, "Transformation-grounded image generation network for novel 3d view synthesis," in *CVPR*, 2017.

[14] Quan Hoang Trung Le, Hung Vu, Tu Dinh Nguyen, Hung Bui, and Dinh Phung, "Learning generative adversarial networks from multiple data sources," in *IJCAI*, 2019, pp. 2823–2829.

[15] Bin Ren, Hao Tang, and Nicu Sebe, "Cascaded cross mlp-mixer gans for cross-view image translation," in *BMVC*, 2021.

[16] Hao Tang, Dan Xu, Nicu Sebe, Yanzhi Wang, Jason J Corso, and Yan Yan, "Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation," in *CVPR*, 2019, pp. 2417–2426.

[17] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al., "Mlp-mixer: An all-mlp architecture for vision," *NeurIPS*, vol. 34, 2021.

[18] Jiachen Li, Ali Hassani, Steven Walton, and Humphrey Shi, "Convmlp: Hierarchical convolutional mlps for vision," *arXiv preprint arXiv:2109.04454*, 2021.

[19] Dan Hendrycks and Kevin Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016.

[20] Justin Johnson, Alexandre Alahi, and Li Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *ECCV*, 2016.

[21] Aliaksandr Siarohin, Enver Sanguineto, Stéphane Lathuiliere, and Nicu Sebe, "Deformable gans for pose-based human image generation," in *CVPR*, 2018, pp. 3408–3416.

[22] Nam N Vo and James Hays, "Localizing and orienting street views using overhead imagery," in *ECCV*. Springer, 2016, pp. 494–509.

[23] Scott Workman, Richard Souvenir, and Nathan Jacobs, "Wide-area image geolocalization with aerial reference imagery," in *ICCV*, 2015, pp. 3961–3969.

[24] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al., "Pytorch: An imperative style, high-performance deep learning library," in *NeurIPS*, 2019.

[26] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017, pp. 1125–1134.