GANZZLE: REFRAMING JIGSAW PUZZLE SOLVING AS A RETRIEVAL TASK USING A GENERATIVE MENTAL IMAGE

Davide Talon[†,‡], *Alessio Del Bue*[†], *Stuart James*[†]

[†]Pattern Analysis and Computer Vision (PAVIS), Istituto Italiano di Tecnologia (IIT), Italy
[‡]Università degli studi di Genova, Italy

ABSTRACT

Puzzle solving is a combinatorial challenge due to the difficulty of matching adjacent pieces. Instead, we infer a mental image from all pieces, which a given piece can then be matched against avoiding the combinatorial explosion. Exploiting advancements in Generative Adversarial methods, we learn how to reconstruct the image given a set of unordered pieces, allowing the model to learn a joint embedding space to match an encoding of each piece to the cropped layer of the generator. Therefore we frame the problem as a $R@1$ retrieval task, and then solve the linear assignment using differentiable Hungarian attention, making the process end-to-end. In doing so our model is puzzle size agnostic, in contrast to prior deep learning methods which are single size. We evaluate on two new large-scale datasets, where our model is on par with deep learning methods, while generalizing to multiple puzzle sizes.

***Index Terms*—** Jigsaw puzzle, Generative Adversarial Network, Hungarian Algorithm, Spatial Reasoning

1. INTRODUCTION

Historically, jigsaw puzzles were introduced by the British cartographer John Spilsbury in 1760 as a children's game to develop cognitive reasoning. Since then, the computational problem of solving a puzzle has found several applications such as image reconstruction [1], assembling of broken objects [2, 3], molecular docking [4], and fresco reconstruction [5]. Recently, the visual problem of solving a puzzle from its unordered pieces has attracted a revamped interest in the Machine Learning and Computer Vision communities [6, 7, 8] as an example of a spatial reasoning task that is comparatively easy for humans.

Unfortunately, despite its interest and importance, the Jigsaw puzzle problem is computationally intractable [9]. Hence, image reassembly methods require the development of heuristics to efficiently place the pieces by taking into account not only visual information such as texture, color, but also the geometrical information such as the shape and orientation of each piece. Such features are used to compute pairwise affinities between pieces [7, 10] for a greedy piece placer optimization [11]. Alternatively, the current application of neural networks have several drawbacks which limit their applicability including ability to handle different sized puzzles and lack of global information [12, 13, 14].

In contrast, we aim to find a global solution by first generating the whole image from the pieces and then estimating their positions. Specifically, we propose GANzzle including a Generative Adversarial Network (GAN) based on a multi-encoder single-decoder architecture, which can then place pieces by evaluating the matching of segments and target slots in the generated image, using a differentiable hungarian attention (Fig. 1). The contributions are therefore threefold: 1) A many-to-one GAN Architecture for recovering a global image from its pieces; 2) Dynamic puzzle size solver using Hungarian attention and contrastive loss; 3) Two new large-scale puzzle solving datasets, named *PuzzleCelebA* and *PuzzleWikiArts*.

2. RELATED WORK

Two main computational approaches have been explored to solve the image jigsaw problem optimization and deep learning methods.

Optimization: exploit edges of pieces to formulate a compatibility metric which can then be optimized. Cho et al. [6] formulate the puzzle-solving problem through a graphical model for label assignment to slots, then the belief propagation optimization shares neighbor information to already placed pieces. By considering the continuity of gradients in adjacent pieces, [15] casts puzzles as a minimum spanning tree problem where edges represent spatial relationships between pieces. To overcome seed sensitivity of the aforementioned methods, [11] iteratively refines the seed to be the largest found segment. Placement and segmentation methods are based on the best-buddy heuristic where pieces agree on being neighbors. Jointly with a relative placement of pieces, [16] proposes that the first placed piece should be distinctive and in a distinctive area. In contrast, we propose a global solution that can support local piece placements and therefore

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 964854.

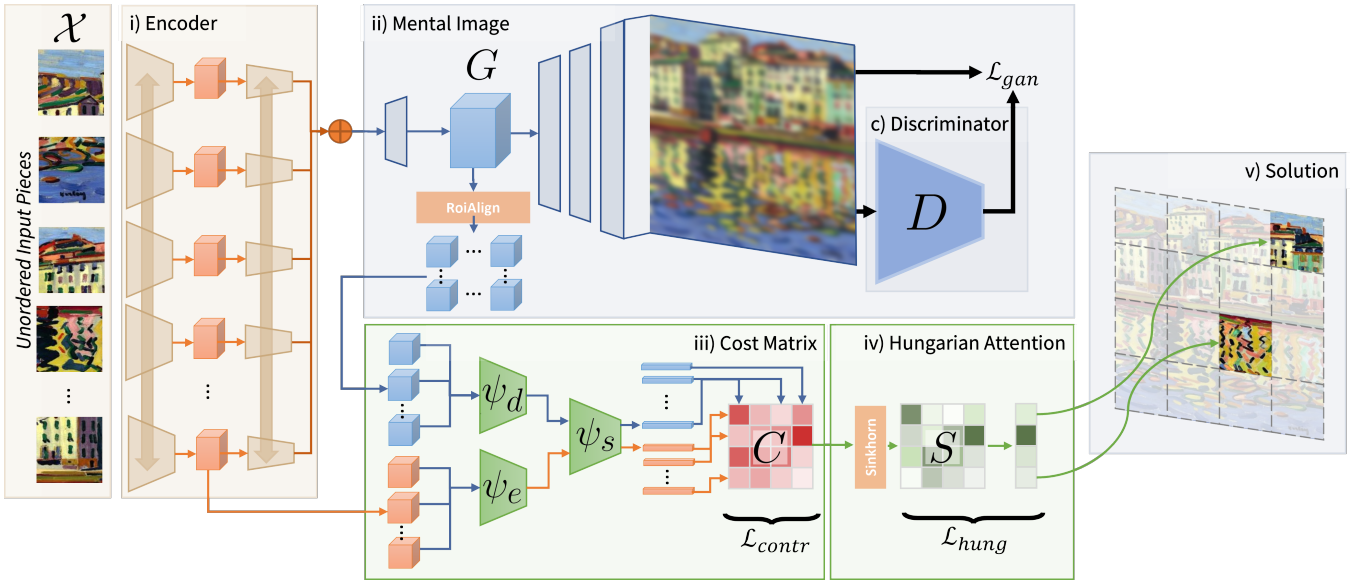

Fig. 1: GANzzle takes as input a set of unordered pieces ($\mathcal{X}$), which is passed through an encoder (i) and then pooled to produce a latent vector. The latent vector is used to generate the *mental image* (ii) using the generator (G) and fine-tuned using a discriminator (D). An intermediate encoding layer of the generator is cropped using RoiAlign to act as targets. (iii) Both the cropped targets and pieces are encoded through embedding networks (ψ_d and ψ_e) and then with a common encoder ψ_s. To produce a cost matrix (C) dot product is used. Hungarian attention (iv) is used to solve for the final location where C is normalized through the iterative Sinkhorn normalization (S) to obtain a doubly stochastic matrix and in turn solve the assignment problem producing the permutation of the solution (v).

be performed efficiently in a single forward pass of a neural network without iterative pair-wise refinement.

Deep Learning: Similar to optimization methods Zhang et al. [17] considered a learnable cost function. An alternating optimization infers the correct permutation and a suitable cost matrix assessing pairwise relationships. The generative model in [14] outputs a placement vector, i.e. a vector whose elements indicate where pieces should be placed, while the discriminator outputs the probability of that vector representing a real-placement. In contrast, we pose a synthesized image is a simpler solution for solving for the piece assignment. [12] tries to infill using a GAN between pairs of pieces and therefore solving the assembly problem. While this may seem similar to GANzzle, its solution is pair-wise and takes $\sim 60s$ to solve a puzzle of 70 pieces, considerably slower than the proposed approach. The regression of the applied permutation matrix, which is intrinsically discrete and hence a non-differentiable problem, is relaxed in [8]. At inference time, the closest permutation matrix is selected via linear integer programming. In Deepzzle [18], a convolutional network predicts relative placement of pieces with respect to a central anchor. By assuming the independent placement of consecutive pieces, the algorithm builds an assembly tree where edges encode log probabilities of placement. [19] employs a generative adversarial network to include semantic information in the placement of pieces, where a classification branch predicts which permutation out of a fixed set has been applied to produce the associated flow-based warp on features. While in [19] the adversarial branch aids the classification of the given permutation by projecting it in the image space, in GANzzle, the generator is learning to permute pieces correctly. As a benefit, our solution can cope with an arbitrary permutation of pieces.

3. PIECE ASSIGNMENT WITH GLOBAL INFORMATION

The GANzzle model takes as input a set of n pieces (or image patches) $\mathcal{X} = \{X_0, X_1, \ldots, X_n\}$, $X_i \in \mathbb{R}^{pw \times ph}, i = 1, \ldots, n$ and aims to infer piece locations supported by the reconstructed target image $\mathcal{Y} \in \mathbb{R}^{tw \times th}$, with p and t being the piece and target dimensions respectively. At first, for each input piece X_i, we learn an embedding representation, then a pooling strategy outputs a fixed-sized encoding which is fed to the decoder to generate a synthetic target image (fig. 1i & fig. 1ii, sec. 3.1). To solve for piece positions, we learn to match the pieces to targets within the global encoding. In the feature space, a cost matrix between patches and target slots accounts for the cost of their assignment (fig. 1iii). Therefore, we optimize the assignments using the differentiable Hungarian attention algorithm which makes the model attend only relevant assignment information (fig. 1iv, sec. 3.2).

3.1. Learning global side information

To create the mental image, i.e., a rough estimate of the global solution, we use an encoder-decoder architecture to generate an image from the pieces. We modify traditional network to accept n pieces as inputs, similar to multi-view approaches [20], where each piece is passed through the multi-encoder which uses shared weights across the pieces. Contrary to the fixed-order assumption of [20], the unordered nature of jigsaw puzzles requires learning piece embeddings and gather them through average-pooling to generate a single encoding vector across all the pieces. The pooled encoding is then passed to the decoder. As with prior work, we found the discriminator's feedback increased high-frequency information, creating more representative synthetic images necessary to guide the subsequent matching of pieces. We opt for MSG-GAN style of approach to provide supervision to the GAN at multiple levels. The generator $G(\cdot)$ and the discriminator $D(\cdot)$ are trained in the standard min max fashion:

$$\mathcal{L}_{gan}(G, D) = \mathbb{E}_y[\log\, D(y)] + \mathbb{E}_x[\log\, (1 - D(G(x)))], \quad (1)$$

where the first and the second expectation are computed on target images $\mathcal{Y}$ and images generated from pieces $\mathcal{X}$, respectively. While the discriminator maximizes eq. (1), the generator minimizes it. As in MSG-GAN, the equations are extended over all resolutions to provide multi-scale gradients:

$$\mathcal{L}_{GAN} = \min_G \max_D \sum_{l=1}^{L} L_{gan}(G^l, D^l) + \lambda_p \mathcal{L}_{mse}(G^l),$$

where G^l is the RGB-converted intermediate representation of the generator at layer l (depth L), D^l the corresponding discriminator. A pixel-wise mean squared error term $\mathcal{L}_{mse}(\cdot)$ is included weighted by λ_p, balancing the reconstruction term with the refinements introduced by the discriminator.

Using MSG-GAN removes the expectation of generating larger images than the sum of input dimension pieces. We exploit this to incrementally grow the GAN during training. Furthermore, we group training into batches based on the jigsaw complexity ($2 \times 2, 3 \times 3, ..., n \times n$). Hence, the network does not need to handle a dynamic number of pieces, while remaining puzzle size agnostic at test time.

3.2. Piece assignment

From the pieces $\mathcal{X}$, we construct a similarity matrix between an intermediate encoding of X_i and the placement target slots of the intermediate embedding of the decoder partitioned by RoiAlign. Both intermediate encodings are passed through shallow networks ψ_e and ψ_d for piece and target slots respectively, to avoid degradation of the GAN. Then, they are jointly embedded through ψ_s to align the embedding spaces. Hence, the similarity matrix is computed as dot product of all possible piece-slot pairs at runtime, making it dynamic to the size of the puzzle. A contrastive loss enforces the feature space to have similar embeddings for piece-slot correct pairs while pushing apart non-corresponding pairs:

$$\mathcal{L}_{contr} = -\mathbb{E}_i \left[\log \frac{exp\left(\psi_s^i \cdot \psi_s^j / \tau\right)}{exp(\psi_s^i \cdot \psi_s^j / \tau) + \sum_{k \neq j} exp(\psi_s^i \cdot \psi_s^k / \tau)} \right]$$

with ψ_s^i and ψ_s^j embeddings of considered piece i and its corresponding slot j, τ the temperature parameter.

Assignments based on the cost matrix could then be efficiently computed by employing the Hungarian Algorithm [21]. However, the algorithm is non-differentiable due to the discrete nature of assignments. Therefore, we employ Hungarian attention (HA) [22] to learn the assignment task in a supervised way, where a continuously relaxed assignment problem is shaped such that its Hungarian solution is the desired one. Initially, the cost matrix (C) is normalized through the iterative Sinkhorn normalization to obtain a doubly stochastic matrix:

$$S^0(C) = \exp(C) \quad (2)$$

$$S^l(C) = F_c\left(F_r\left(S^{l-1}\right)\right) \quad (3)$$

$$S(C) = \lim_{l \to \infty} S^l\left(C\right), \quad (4)$$

where F_c and F_r are the row and column-wise normalization $F_c(C) = C \oslash \left(\mathbf{1}_N \mathbf{1}_N^T C\right)$ and $F_r(C) = C \oslash \left(C \mathbf{1}_N \mathbf{1}_N^T\right)$ respectively, with $\oslash$ denoting the element-wise division and 1_N the n-dimensional unit column vector. Therefore, its assignment counterpart is determined via HA matching $\mathrm{Hung}\,(S)$. Thus, a hard attention mask is generated by comparing it to the ground-truth assignment $\mathbf{S}^G$ through a element-wise logic-OR operator:

$$\mathbf{Z} = OR\left(\mathrm{Hung}\,(\mathbf{S}), \mathbf{S}^G\right).$$

Note that the hard mask catches most relevant elements in the matrix to avoid overconfidence, however, both correct and misplaced pieces are modeled.

The binary cross-entropy loss with respect to the ground-truth assignment matrix is attended through the mask:

$$\mathcal{L}_{hung} = \sum_{i,j \in [n]} \mathbf{Z}_{ij} \left(\mathbf{S}_{ij}^G \log \mathbf{S}_{ij} + \left(1 - \mathbf{S}_{ij}^G\right) \log\left(1 - \mathbf{S}_{ij}\right)\right),$$

where $[n]$ is the set of indexes from 1 to n.

By optimizing the above permutation loss, our model learns to correctly match the Hungarian's assignment computed from S to the ground truth permutation. At inference time, the estimated assignment is hence the Hungarian binarization of the doubly stochastic matrix $\mathrm{Hung}\,(S)$.

The complete loss for the GANzzle model is therefore:

$$\mathcal{L} = \mathcal{L}_{GAN} + \mathcal{L}_{hung} + \mathcal{L}_{contr}. \quad (5)$$

We additionally consider a model that exploits only HA without the generated mental image as permutation based method referred to Hung-perm (See supp. mat. for full details).

Dataset	PuzzleCelebA				PuzzleWikiArts			
	6x6	8x8	10x10	12x12	6x6	8x8	10x10	12x12
Paikin and Tal [16]	99.12	98.67	98.39	96.51	98.03	97.35	95.31	90.52
Pomeranz et al. [11]	84.59	79.43	74.80	66.43	79.23	72.64	67.70	62.13
Gallagher [15]	98.55	97.04	95.49	93.13	88.77	82.28	77.17	73.40
PO-LA [17]	71.96	50.12	38.05	-	12.19	5.77	3.28	-
Hung-perm	33.11	12.89	4.14	2.18	8.42	3.22	1.90	1.25
GANzzle-Single (Ours)	71.00	51.81	**43.74**	-	11.78	6.23	**8.97**	-
GANzzle (Ours)	**72.18**	**53.26**	32.84	**12.94**	**13.48**	**6.93**	4.10	**2.58**

Table 1: Results for direct comparison accuracy on PuzzleCelebA and PuzzleWikiArts. We directly compare against deep method ([17]) and without mental image (Hung-perm) for similar computational performance and include optimization methods [16, 11, 15] for complete comparison. We note, that in contrast to GANzzle, [17] is trained one model per size.

Model	Missing (%)			Noisy (σ)			Eroded (px)		
	10%	20%	30%	0.05	0.1	0.2	1	2	5
Paikin and Tal [16]	-	-	-	51.51	7.73	3.31	2.82	2.77	2.79
Pomeranz et al. [11]	52.43	24.26	25.99	87.84	89.63	91.50	6.01	16.30	15.15
Gallagher [15]	79.68	66.02	51.17	96.39	98.34	97.75	32.55	18.59	6.27
PO-LA [17]	**64.35**	**60.10**	**58.60**	**69.87**	**65.30**	**49.85**	23.81	10.93	4.84
Hung-perm	29.79	26.45	23.88	31.84	29.01	21.45	25.45	26.01	**9.50**
GANzzle-Single	59.00	46.24	37.54	63.81	48.95	10.13	**28.75**	**39.21**	9.23
GANzzle	58.50	44.70	35.01	64.51	37.72	6.81	28.59	35.47	4.70

Table 2: Comparison of missing pieces (except [16]), Gaussian noise and eroded pieces on a 6×6 puzzle for PuzzleCelebA.

4. EVALUATION

We evaluate on the PuzzleCelebA ($30k$ images) and PuzzleWikiArts ($63k$ images) datasets derived from their respective CelebA [23] and part of WikiArts [24]. We provide train and testing split ($80-20\%$) and the permutation for puzzle sizes $[2, 4, 6, 8, 10, 12]$. For evaluation, we use standard **direct comparison metric** [6] where an assignment is correct if it is placed in the correct position. Full details of the datasets and extended evaluation (inc. Neighbor comparison metric) can be found in the supplementary material.[1]

We show Direct Accuracy for both datasets in Table 1, for optimization methods [16, 11, 15], deep method [17] and based only on patch embedding and Hungarian attention (Hung-perm). For PO-LA, as it is fixed size we train one model per size, however, 12×12 was limited due to memory, we include the single size GANzzle(-Single) for direct comparison. Our method generalizes across sizes with similar performance to the single version up to 10×10. In comparison, GANzzle is competitive with [17] and out performs Hung-perm on large sizes benefiting from the mental image.

In addition, we evaluate on PuzzleCelebA with missing, noisy, and eroded (missing border) pieces for 6×6 puzzles in Table 2. For missing pieces we are competitive against [17] and [11]. However, GANzzle struggles in contrast to other methods with additive noise, although it outperforms Hung-Perm benefiting from the mental image. While for eroded pieces GANzzle outperforms [17] and Hung-Perm, in addition for small erosion (1&2px) are competitive with optimization methods. GANzzle consistently performs similar to the single size version highlighting the generalization. Limitations emerge with challenging pieces, i.e., pieces with similar content, as they can be swapped. The qualitative analysis in Fig. 2 shows how GANzzle is able to resolve for the structure.

Fig. 2: Qualitative results of GANzzle for 10×10 on PuzzleWikiArts and PuzzleCelebA (see Supp.).

5. CONCLUSION

Our global to local solution GANzzle is able to learn to position jigsaw puzzle pieces correctly with a single forward pass of the network. The use of Hungarian attention is dynamic to jigsaw puzzle sizes and successfully solves for the location of pieces using a singular model. We have shown results on two new datasets to standardize the jigsaw puzzle solving problem, where we perform competitively with deep methods but overcome the single-size model problem.

[1] Supp. Mat. is available at https://github.com/IIT-PAVIS/GANzzle

6. REFERENCES

[1] A. van den Hengel, C. Russell, A. Dick, J. Bastian, D. Pooley, L. Fleming, and L. Agapito, "Part-based modelling of compound scenes from images," in *CVPR*, 2015, pp. 878–886.

[2] G. Palmas, N. Pietroni, P. Cignoni, and R. Scopigno, "A computer-assisted constraint-based system for assembling fragmented objects," in *Digital Heritage International Congress*, 2013, vol. 1, pp. 529–536.

[3] Hanan ElNaghy and Leo Dorst, "Complementarity-preserving fracture morphology for archaeological fragments," in *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer, 2019, pp. 403–414.

[4] Miguel L Teodoro, George N Phillips, and Lydia E Kavraki, "Molecular docking: A problem with thousands of degrees of freedom," in *ICRA*. IEEE, 2001, vol. 1, pp. 960–965.

[5] Benedict J. Brown, Corey Toler-Franklin, Diego Nehab, Michael Burns, David Dobkin, Andreas Vlachopoulos, Christos Doumas, Szymon Rusinkiewicz, and Tim Weyrich, "A system for high-volume acquisition and matching of fresco fragments: Reassembling theran wall paintings," 2008, SIGGRAPH.

[6] T. S. Cho, S. Avidan, and W. T. Freeman, "A probabilistic image jigsaw puzzle solver," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 183–190.

[7] D. Sholomon, O. David, and N. S. Netanyahu, "A genetic algorithm-based solver for very large jigsaw puzzles," in *CVPR*, June 2013, pp. 1767–1774.

[8] Rodrigo Santa Cruz, Basura Fernando, Anoop Cherian, and Stephen Gould, "Visual permutation learning," in *CVPR*, 2017.

[9] Erik D. Demaine and Martin L. Demaine, "Jigsaw puzzles, edge matching, and polyomino packing: Connections and complexity," *Graphs and Combinatorics*, vol. 23, no. 1, pp. 195–208, Jun 2007.

[10] K. Son, J. Hays, and D. B. Cooper, "Solving square jigsaw puzzle by hierarchical loop constraints," *IEEE PAMI*, pp. 1–1, 2018.

[11] D. Pomeranz, M. Shemesh, and O. Ben-Shahar, "A fully automated greedy square jigsaw puzzle solver," in *CVPR 2011*, June 2011, pp. 9–16.

[12] Dov Bridger, Dov Danon, and Ayellet Tal, "Solving jigsaw puzzles with eroded boundaries," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3526–3535.

[13] Mehdi Noroozi and Paolo Favaro, "Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles," in *ECCV 2016*, Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, Eds., Cham, 2016, pp. 69–84, Springer International Publishing.

[14] A. Rafique, T. Iftikhar, and N. Khan, "Adversarial placement vector learning," in *ICACS*, 2019, pp. 1–7.

[15] A. C. Gallagher, "Jigsaw puzzles with pieces of unknown orientation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 382–389.

[16] Genady Paikin and Ayellet Tal, "Solving multiple square jigsaw puzzles with missing pieces," *IEEE CVPR*, pp. 4832–4839, 2015.

[17] Yan Zhang, Jonathon Hare, and Adam Prügel-Bennett, "Learning representations of sets through optimized permutations," in *ICLR*, 2019.

[18] Marie-Morgane Paumard, David Picard, and Hedi Tabia, "Deepzzle: Solving visual jigsaw puzzles with deep learning and shortest path optimization," *IEEE Transactions on Image Processing*, vol. 29, pp. 3569–3581, 2020.

[19] Ru Li, Shuaicheng Liu, Guangfu Wang, Guanghui Liu, and Bing Zeng, "Jigsawgan: Self-supervised learning for solving jigsaw puzzles with generative adversarial networks," *arXiv preprint arXiv:2101.07555*, 2021.

[20] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik G. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proc. ICCV*, 2015.

[21] Harold W Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[22] Tianshu Yu, Runzhong Wang, Junchi Yan, and Baoxin Li, "Learning deep graph matching with channel-independent embedding and hungarian attention," in *International conference on learning representations*, 2019.

[23] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[24] W. R. Tan, C. S. Chan, H. E. Aguirre, and K. Tanaka, "Improved artgan for conditional synthesis of natural image and artwork," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 394–409, Jan 2019.