Proceedings of the 2nd International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/76/20240579

A review of recent image synthesis techniques

Chengwei Hu

Wuhan University, Luojia Hill, Wuchang District, Wuhan, Hubei, 430072, China

263200693@qq.com

Abstract. Deep neural networks typically demand a substantial volume of labeled training data. Nonetheless, the process of gathering and annotating this training data is frequently labor-intensive and time-consuming. One way to address this issue is through Image Composition. Image composition refers to the process of combining multiple visual elements or components. It may involve techniques like image synthesis or generation, where algorithms generate new images by combining elements from existing ones.

This paper aims to overview the recent advancements in Image Composition based on deep networks and relevant applications. It offers an extensive overview of the concepts, principles, and techniques related to Image Composition. Subsequently, it delves into two distinct levels at which image composition can be accomplished: specifically, with 2D elements and 3D objects. Furthermore, the method of achieving composition with point clouds is presented.

Keywords: Image Composition, Deep Learning, 3D Image Composition, Point Cloud, Point Cloud Composition.

1. Introduction

Image composition has been extensively explored within the computer graphics research community, involving intricate devices and heuristic techniques. It has garnered increased attention in the era of deep learning.

Image composition through synthesis aims to integrate foreground objects into background images while detecting plagiarism, meeting three essential criteria:

- Strategic Placement: Ensuring semantic coherence is essential. Foreground objects should be positioned in a way that makes sense within the context.
- Authentic Appearance: Ensuring that embedded foreground objects blend seamlessly with the background image requires consistency in color, brightness, and texture.
- Geometric Realism: When embedding objects of interest, maintain accurate perspective and geometric alignment with the surrounding structures. [1]

Considering these criteria, researchers in the academic community have put forth various image synthesis strategies. These approaches can be broadly classified into three main types: 2D Image Composition, 3D Image Composition, and Point Cloud Composition.

2D Image Composition involves creating a novel image by combining elements from two or more 2-dimensional images, adhering to human visual conventions. 3D Image Composition involves integrating three-dimensional objects into a scene to create a new, depth-enhanced, realistic composite image.

DOI: 10.54254/2755-2721/76/20240579

Point Cloud Composition involves the process of combining and integrating point cloud data into a cohesive and realistic scene.

2. 2D image composition

2.1. SynthText

Ankush and colleagues proposed the SynthText algorithm. SyntheText was produced using an efficient and scalable engine designed for generating synthetic images containing text within cluttered scenes. [2] In the context of SyntheText synthesis, a dense pixel-wise depth map of the background image is obtained using CNNs. Next, the background image is divided into connected regions based on local color and texture. For each of these contiguous regions, an estimate of the local surface normal is computed. Subsequently, an appropriate text color is chosen based on the region's color. Finally, a text sample is generated using a randomly selected font and adjusted according to the local surface orientation. The seamless integration of the text into the scene is achieved using Poisson image editing. The SynthText dataset comprises foreground text and background images, with the foreground text sourced from the Newsgroup 20 dataset, including words, text lines, or paragraphs. As for the background images (8000), they are obtained from Google Image Search and must not contain any embedded text. Remarkably, the trained model using 800K SynthText images can even surpass state-of-the-art (SOTA) models trained on natural scene text images. During inference, FCRN multi-scale leverages multiple scales, while FCRN + multi-filt incorporates post-processing techniques such as box regression and non-maximum suppression (NMS).

2.2. Verisimilar

In 2018, Fangneng and colleagues introduced an innovative image synthesis technique that utilizes semantic segmentation maps, visual saliency, and realistic text color and brightness. This approach enables the rapid generation of a substantial number of annotated images, which can be used for training accurate and robust scene text detection and recognition models.

The input data includes "Background Images" and "Source Texts". [3] Semantic coherence (SC) ensures that text within a scene is positioned in semantically meaningful regions of the background image.

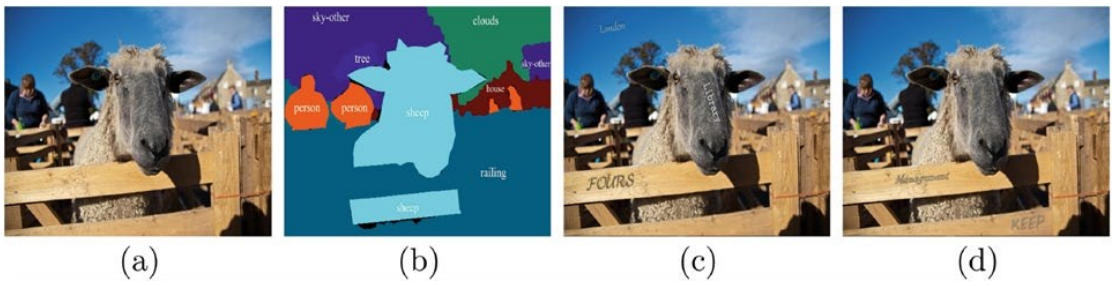

Figure 1. Semantic Coherence (SC) in image synthesis: (a) represents the original image. In the context of semantic segmentation, the absence of semantic coherence (SC) can lead to texts being embedded in arbitrary regions—such as the sky or the head of a sheep—which are rarely observed in typical scenes (as depicted in (c)). SC ensures that texts are placed in semantically meaningful regions, as shown in (d). It achieves this coherence by leveraging ground-truth annotations from semantic segmentation datasets.

In practical scenarios, scene text is typically positioned near homogeneous regions, such as signboards, to enhance contrast and visibility. This contrasts with situations where text is split between the surface of a machine and the sky. Leveraging such observations, visual saliency serves as a guide for determining precise scene text embedding locations.

DOI: 10.54254/2755-2721/76/20240579

During image processing, the saliency model generates a saliency map. In practical scenarios, homogeneous regions typically exhibit lower saliency compared to highly contrasting or cluttered areas. Consequently, positions with lower saliency are selected for embedding scene text. Achieving this can be facilitated by defining a global threshold.

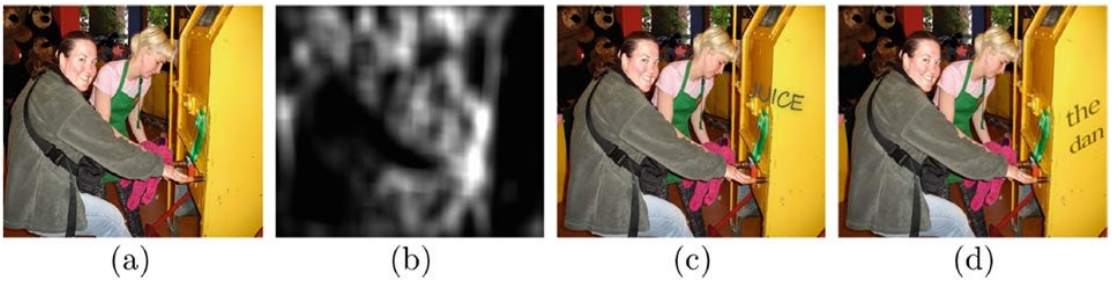

Figure 2. In the context of image synthesis guided by Saliency Guidance (SG), consider the following: (a) represents the original image. (b) displays the saliency map derived from the image. In the absence of SG, texts may be embedded across object boundaries, as exemplified in (c). However, such placements are rarely observed in typical scenes. SG ensures that texts are appropriately embedded within semantically meaningful regions, as depicted in (d).

Adaptive text appearance adapts its color, contrast, and orientation based on real-world scenes. This concept entails locating patches of scene text images that are easily accessible in existing datasets. These patches exhibit backgrounds with colors and brightness akin to the previously determined background areas. By analyzing the color and brightness of pixels within these identified patches, the color and brightness of the original texts can be determined.

2.3. SF-GAN

Fangneng et al. (2019) proposed a novel Spatial Fusion Generative Adversarial Network (SF-GAN). SF-GAN strives to seamlessly incorporate objects of interest into background images, ensuring realism by managing factors such as object size, perspective, and lighting. [4]

The SF-GAN consists of both a geometry synthesizer and an appearance synthesizer, creating a network that can be trained end-to-end.

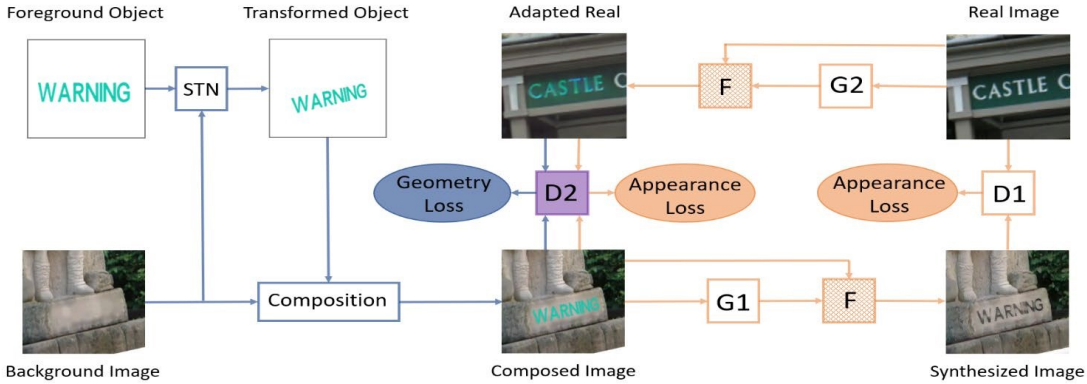

Figure 3. SF-GAN comprises two main components: The Geometry Synthesizer, emphasized by blue markings on the left, plays a pivotal role, while the Appearance Synthesizer, highlighted by orange markings on the right, incorporates guided filters, G1, G2, D1, and D2.

The geometry synthesizer focuses on understanding the local geometry of background images. This knowledge allows it to transform and seamlessly place foreground objects into those backgrounds. The geometry synthesizer comprises three main components: a spatial transform network (STN), a

DOI: 10.54254/2755-2721/76/20240579

composition module, and a discriminator. STN takes the concatenated foreground object and background image as input. The modified foreground element can subsequently blend into the background picture, forming an initially combined setting. The discriminator D2 is trained to discern the realism of this integrated image when compared against a collection of genuine images.

For effective training of the geometry synthesizer, it is crucial that the reference images exhibit realism within the geometry space. Simultaneously, they should exhibit similar appearance characteristics to the initially composed images. It achieves this by utilizing Adapted Real images from the appearance synthesizer as the reference during geometry synthesizer training. D1 is designed to distinguish between the adjusted composed images (i.e., the Composed Image following domain adaptation by G1) and authentic images. This procedure prompts G1 to understand how to transition from the Composed Image to Final Synthesis images that exhibit realism in the appearance domain. Meanwhile, G2 is tasked with learning the mapping from Real Images to Adapted Real images.

When performing image-to-image translation, the Composed Image undergoes a process that can inadvertently smooth out certain image details. However, by utilizing the guided filter F, the Translated Image (which corresponds to the output of G1) can better preserve full details in both the background face and the foreground hat areas, resulting in a more faithful representation of the original scene.

Mixup [5] involves training a neural network by employing combinations of pairs of instances and their associated labels, ensuring convexity in the combinations. By doing so, mixup effectively regularizes the neural network, favoring a simpler linear behavior between training examples.

$$X_{mix} = \lambda \cdot X + (1 - \lambda) \cdot X'$$

$$Y_{mix} = \lambda \cdot Y + (1 - \lambda) \cdot Y'$$

In this context, (X; Y) and $(X'; Y')$ represent two instances from the training dataset, with λ indicating a scalar value within the range [0, 1].

CutMix [6] is a technique that replaces a region within an image with a patch from another image. Furthermore, the ground truth labels are blended proportionally, taking into account the number of pixels in the combined images. This approach aims to reduce the likelihood of plagiarism detection when applied to combined images.

$$X_{mix} = M \odot x_i + (1 - M)x_j$$

$$Y_{mix} = \lambda y_j + (1 - \lambda)y_j$$

In this scenario, $M \in \{0,1\}^{W \times H}$ represents a binary mask denoting the specific region within the image intended for mixing.

AugMix [7] is a data processing technique that enhances model robustness against unforeseen data shifts, particularly when the training distribution differs from the test distribution. It introduces randomness and diversity, enhancing the model's capacity to adapt to a wide range of uncertainties and data perturbations.

3. 3D image composition

While numerous prior studies concentrate on managing objects in the foreground of 2D images, it's important to recognize that foreground elements in 3D models provide increased flexibility due to their unrestricted 360-degree viewing capability.

3.1. VA-GAN

View Alignment GAN (VA-GAN) addresses the limitations of 2D foreground object composition, particularly in scenarios with restricted views. By utilizing both 3D models and 2D background images, VA-GAN forecasts authentic views and orientations for the 3D models in alignment with the background image. Moreover, it produces cohesive textures for objects, facilitating a seamless integration of 3D models into 2D images with a remarkable level of realism.[8]

DOI: 10.54254/2755-2721/76/20240579

From the background image Background1, the View Generator anticipates a View Code using transformation parameters derived from the local geometry of Background1. This guidance process involves a discriminator Dv, which utilizes genuine pairs of object-background to acquire knowledge on geometric alterations. Following this: The Projector produces a depth map for the 3D model based on the View Code. Subsequently, the Texture Generator maps the depth map to authentic textures for the 3D model.

However, VA-GAN excels in achieving geometry alignment and realistic texture generation. The images VA-GAN1, VA-GAN2, and VA-GAN3 are created utilizing identical 3D models and style codes, albeit with different background images.

3.2. EMLight

EMLight introduces a spherical distribution approximation that estimates the background image's illumination and applies it to render the embedded 3D object. The core of EMLight is a regression network that learns to estimate lighting parameters based on a given 2D image and its corresponding illumination map.

It breaks down the illumination map into three elements: light distribution, luminance, and ambient factors. These elements correspondingly signify the distribution and intensity of light sources, as well as the average energy excluding the light sources. Hence, the task of predicting illumination is reframed as a regression problem for the distribution, intensity, and ambient characteristics of light.

For regressing light distributions, a metric called the Spherical Mover's Loss (SML) has been devised. SML gauges the distance between two spherical distributions by identifying the minimum radian distance required to shift one distribution onto another across the sphere's surface.

Moreover, it integrates a neural projector tasked with acquiring an estimation of an illumination map for the 2D image. This estimation relies on the illumination parameters forecasted by the aforementioned regression network. As a result, the embedded 3D object can be efficiently rendered utilizing the estimated illumination map. [9]

4. Point Cloud Composition

LiDAR (Light Detection and Ranging) point clouds excel at accurately capturing 3D information and remain robust even in the face of illumination variations. However, annotating point clouds is an extremely labor-intensive process.

To address this challenge, PointMixup enhances instance-level point clouds by performing interpolation between samples [10].

The aim is to create a mechanism capable of acquiring knowledge on mapping a point cloud with a variety of semantic labels.

In a metric space (S, d), a shortest-path interpolation refers to an interpolation linking the provided pair of source examples $S_1 \in S$ and $S_2 \in S$. For any λ within the interval [0, 1], the equation $d\left(S_1, S^{(\lambda)}\right) + d\left(S^{(\lambda)}, S_2\right) = d(S_1, S_2)$ holds, with $S^{(\lambda)}$ representing the interpolant.

When working with point clouds, a suitable metric for measuring distance is the Earth Mover's Distance (EMD). EMD calculates the minimum displacement required for each point in the first point cloud, denoted as x_i in S_1, to match with a corresponding point in the second point cloud, denoted as y_j in S_2. The use of EMD in point clouds addresses the subsequent challenge of assigning points effectively:

$$\Phi^* = \arg\min_{\phi \in \Phi} \sum_i \left\| x_i - y_{\phi(i)} \right\|_2$$

After establishing the optimal assignment ϕ^*, the EMD is then defined as the average effort needed to move points from S_1 to S_2:

DOI: 10.54254/2755-2721/76/20240579

$$d_{EMD} = \frac{1}{N}\sum_{i}\left\|x_i - y_{\phi^*(i)}\right\|_2$$

It introduces an interpolation method known as Optimal Assignment (OA) Interpolation. When provided with a pair of source point clouds, S_1 and S_2, the OA interpolation generates the requisite augmented data for mixing point clouds. The procedure can be outlined as per the subsequent algorithm:

$$S_{OA}^{(\lambda)} = \{(1-\lambda)\cdot x_i + \lambda \cdot y_{\phi^*(i)}\}_{i=1}^{N}$$

Jinlai et al. proposed a simple and effective point cloud data augmentation method, named PointCutMix. PointCutMix generates instance-level point clouds by substituting object points in one LiDAR scan with corresponding object points from another LiDAR scan. [11]

Following the method in PointMixup, the assignment function in the EMD is defined as:

$$\phi^* = \arg\min_{\phi\in\Phi}\sum_{i}\left\|x_{1,i} - x_{2,\phi(i)}\right\|_2$$

The EMD is defined as:

$$EMD = \frac{1}{N}\sum_{i}\left\|x_{1,i} - x_{2,\phi^*(i)}\right\|_2$$

Upon securing the optimal assignment ϕ^* between two samples, it characterizes the merging operation as follows:

$$\tilde{x} = B\cdot x_1 + (I_N - B)\cdot \widetilde{x_2}$$

$$\tilde{y} = \lambda y_1 + (1-\lambda)y_2$$

Here, B is defined as the diagonal matrix $diag\{b_1, b_2, \ldots b_n\}$, where each b_i is either 0 or 1, signifying the sample to which the point is assigned.

If b_i is equal to 1, the i^{th} point is chosen from x_1; otherwise, it is replaced by the optimally assigned point in x_2. In denotes an identity matrix.

Two substitution strategies are introduced to construct the diagonal matrix B. The initial approach, known as PointCutMix-R, randomly selects a subset of n points from x_1. These points are assigned a value of 1 in B, signifying that they will remain unchanged. The remaining points are assigned a value of 0. Furthermore, to preserve the local features of the point cloud, it devises a second strategy, termed PointCutMix-K. This approach randomly chooses a central point p from x_1, then identifies its n - 1 closest neighbors. It merges p with its nearest neighbors and designates these points with a value of 1 in B. Correspondingly, the remaining points are assigned a value of 0 and are subject to replacement.

It can be noted that the samples produced by PointCutMix-R seem to intersect like two entities, whereas the amalgamated data from PointCutMix-K distinctly portray the fusion of two object components.

LiDAR-Aug creates point clouds at the scene level by incorporating point-cloud objects from different LiDAR scans into the existing scan. This method is effective in enhancing the representation of infrequent objects and corner cases.

The suggested framework comprises two components, specifically the Pose Generator and the Rendering Module. [12]

PolarMix operates on a broader scale, exchanging sections of point clouds between two LiDAR scans separated along the azimuth axis. It performs a rotation and pasting operation at the instance level, extracting point instances from one LiDAR scan, rotating them, and then integrating these rotated instances into other scans. [13]

DOI: 10.54254/2755-2721/76/20240579

5. Conclusion

This paper provides an overview of recent advancements in image synthesis techniques based on deep learning and their relevant applications. Image composition refers to the process of combining multiple visual elements or components, and it may involve techniques such as image synthesis or generation, where algorithms generate new images by combining elements from existing ones.

In this paper, we have extensively discussed the concepts, principles, and techniques related to image composition. Furthermore, we have delved into two distinct levels at which image composition can be accomplished: specifically, with 2D elements and 3D objects. Additionally, we have introduced methods for achieving composition with point clouds.

In the realm of 2D image composition, algorithms such as SynthText, Verisimilar, and SF-GAN have been introduced, each employing different strategies and technologies to achieve image composition. These methods have made significant progress in ensuring semantic coherence, authentic appearance, and geometric realism between foreground objects and background images.

In the domain of 3D image composition, methods like VA-GAN and EMLight leverage 3D models and 2D background images to achieve a higher degree of realism and fidelity in image composition.

Lastly, in point cloud composition, techniques such as PointMixup, PointCutMix, and LiDAR-Aug enhance instance-level representations of point cloud data, thereby improving model robustness and performance.

In conclusion, image synthesis techniques hold great promise in various fields, and future research will further explore and expand these techniques to meet the growing demands of applications.

References

[1] Li, X., Teng, G., An, P., & Yao, H. (2023). Image Composition Method Based on a Spatial Position Analysis Network. *Electronics*, *12*(21), 4413.

[2] Gupta, A., Vedaldi, A., & Zisserman, A. (2016). Synthetic data for text localisation in natural images. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2315-2324).

[3] Zhan, F., Lu, S., & Xue, C. (2018). Verisimilar image synthesis for accurate detection and recognition of texts in scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 249-266).

[4] Zhan, F., Zhu, H., & Lu, S. (2019). Spatial fusion gan for image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3653-3662).

[5] Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2017). mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

[6] Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., & Yoo, Y. (2019). Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6023-6032).

[7] Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., & Lakshminarayanan, B. (2019). Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*.

[8] Zhang, C., Zhan, F., Lu, S., Ma, F., & Xie, X. (2020). Towards realistic 3d embedding via view alignment. *arXiv preprint arXiv:2007.07066*.

[9] Zhan, F., Zhang, C., Yu, Y., Chang, Y., Lu, S., Ma, F., & Xie, X. (2021, May). Emlight: Lighting estimation via spherical distribution approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 4, pp. 3287-3295).

[10] Chen, Y., Hu, V. T., Gavves, E., Mensink, T., Mettes, P., Yang, P., & Snoek, C. G. (2020). Pointmixup: Augmentation for point clouds. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16* (pp. 330-345). Springer International Publishing.

[11] Zhang, J., Chen, L., Ouyang, B., Liu, B., Zhu, J., Chen, Y., ... & Wu, D. (2022). Pointcutmix: Regularization strategy for point cloud classification. *Neurocomputing*, *505*, 58-67.

DOI: 10.54254/2755-2721/76/20240579

[12] Fang, J., Zuo, X., Zhou, D., Jin, S., Wang, S., & Zhang, L. (2021). Lidar-aug: A general rendering-based augmentation framework for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4710-4720).

[13] Xiao, A., Huang, J., Guan, D., Cui, K., Lu, S., & Shao, L. (2022). Polarmix: A general data augmentation technique for lidar point clouds. *Advances in Neural Information Processing Systems*, *35*, 11035-11048.