Place Anything into Any Video

Ziling Liu[1,2] , **Jinyu Yang**[1*] , **Mingqi Gao**[1] , **Feng Zheng**[1,2]
[1]tapall.ai
[2]Southern University of Science and Technology
{ziling.liu, jinyu.yang, mingqi.gao}@tapall.ai, f.zheng@ieee.org

Abstract

Controllable video editing has demonstrated remarkable potential across diverse applications, particularly in scenarios where capturing or recapturing real-world videos is either impractical or costly. This paper introduces a novel and efficient system named *Place-Anything*, which facilitates the insertion of any object into any video solely based on a picture or text description of the target object or element. The system comprises three modules: 3D generation, video reconstruction, and 3D target insertion. This integrated approach offers an efficient and effective solution for producing and editing high-quality videos by naturally inserting realistic objects. Through experiment, we demonstrate that our system can effortlessly place any object into any video using just a photograph of the object. Our demo video can be found at https://youtu.be/afXqgLLRnTE. Please also visit our project page https://place-anything.github.io to get more information.

1 Introduction

When producing movies or commercials, acquiring video footage can often be a costly endeavor, involving significant expenses for outdoor shoots or the construction of indoor scenes. Once filmed, modifying or editing the video content becomes expensive and time-consuming, which can only be achieved by professional post-production engineers. For non-professionals and non-developers, inserting virtual objects into pre-existing videos to create a smooth visual effect can be an even more challenging task. This is primarily due to the complex operational requirements of certain post-production software and the difficulties associated with obtaining accurate 3D models. Consequently, there is a dire need for a user-friendly interaction solution that can simplify this process and make it more accessible. Moreover, such a solution can be applied to a range of applications including virtual reality, video composition, advertisement insertion, and so on. However, achieving the goal of "placing anything into any video" faces multiple challenges. For real

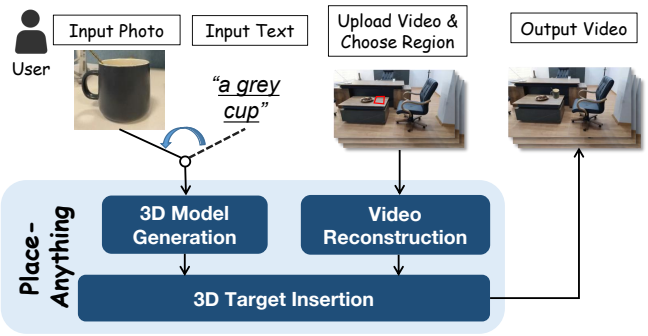

Figure 1: A diagram showing how our *Place-Anything* system works. The system takes source materials from users and generates required videos.

videos, both spatial and temporal consistency are crucial, requiring inserted objects to strictly adhere to the principles of the physical world. Nevertheless, for ease of use, the source materials provided by users should be simple and convenient, such as photos or descriptive narratives, which does not satisfy consistency well. Moreover, for videos that need to be edited, camera parameters are mostly unknown.

As our goal is to develop user-friendly applications, we are exploring whether the "2D $\rightarrow$ 3D $\rightarrow$ 2D" solution can effectively address the aforementioned challenges. To be detailed, the source materials, both photos and videos, are inherently 2D. However, the editing process occurs within a 3D space, ultimately resulting in re-created 2D videos. Therefore, the 3D model generation and insertion become the key to such a solution. Recently, the 3D generation techniques [Poole *et al.*, 2022], [Raj *et al.*, 2023], [Liu *et al.*, 2023] have gained rapid development. However, all methods above can only generate radiance fields that are hard to use in downstream applications like 3D modification and 3D rendering.

Even if a 3D model is obtained, traditional post-production softwares and Augmented Reality (AR) techniques, which can render the model into video, have limitations on real-world applications. Firstly, the motion tracking tool boujou can only reconstruct the sparse point cloud from the tracking key points thus hard to localize the textureless region. Besides, given the results of boujou, it is still necessary to set

∗ Corresponding author.

https://www.vicon.com/resources/blog/faq_tag/boujou/

up the camera and select a 3D plane in other 3D graphical software like 3ds Max, which is hard for non-professionals or non-developers. Some AR SDKs, *e.g.*, ARKit in IOS and ARcore in Android, can render the 3D model into the shooting video stream with the fusion of visual SLAM or auxiliary sensors like IMU or LiDARs. However, traditional SLAM pipelines depend on accurate camera intrinsic to reconstruct the scene and camera poses. Therefore, these AR applications gaining accurate camera intrinsic from mobile devices can not deal with arbitrary video footage offline.

To address the challenges, we introduce the *Place-Anything* system, a workflow encompassing 3D model generation, video reconstruction, and 3D target insertion. As demonstrated in Figure 1, it is based on the 3D generation model and video self-calibration techniques, allowing users to customize their 3D assets and integrate them into any videos, creating intriguing visual effects without prior 3D rendering knowledge. This innovative approach revolutionizes product advertisements and video post-production, allowing users to digitize tangible objects or imaginary concepts with ease. The integration of reality and digital content opens new horizons for creativity and video manipulation. *Place-Anything* dynamically inserts objects into diverse videos, such as advertisements and influencer content, enhancing video producers' editing capabilities and elevating viewer experiences. Thus, *Place-Anything* has at least the following advantages:

1) Versatility: *Place-Anything* boasts remarkable adaptability, allowing various objects and video scenarios. This frees users from the tedium of traditional video production and reproduction, as both tangible items and abstract concepts can be easily transformed into digital assets. Moreover, it is agnostic to the source of videos, whether captured or generated, as it does not rely on specific camera parameters.

2) Interactivity: *Place-Anything* features an intuitive user interface, enabling users to create and customize videos with ease. With just a few clicks, users can select the desired region, adjust the scale and orientation of digital assets based on previews, and naturally integrate them into videos. This ensures precise and effortless video manipulation.

3) High consistency: Unlike text-to-video methods, *Place-Anything* is committed to delivering highly consistent results. Our system generates accurate 3D meshes from photos or text descriptions, ensuring that the digital representations of objects maintain their original quality and details. This consistency extends to the consistent integration of objects into pre-existing videos, resulting in a natural and realistic blend of content. This level of controllability ensures that users can achieve their desired outcomes while maintaining the authenticity and integrity of the original video content.

In summary, *Place-Anything* offers a comprehensive solution for video manipulation, combining versatility, interactivity, and consistency to deliver exceptional results for users. Whether you're a professional video producer or a casual user, our system enables you to smoothly blend reality and digital content, unlocking a world of creative possibilities.

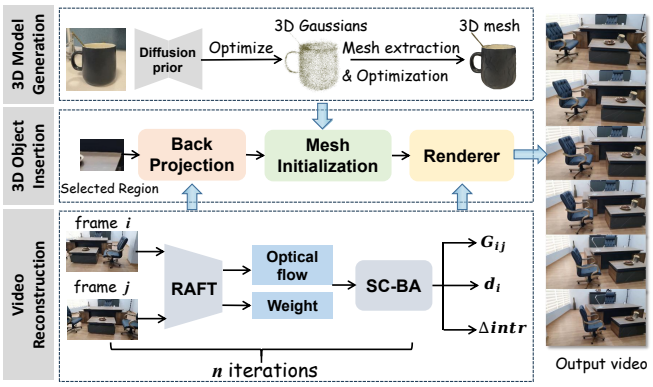

Figure 2: Overview of the pipeline of our *Place-Anything* system.

2 System Architecture

Overview. Figure 2 gives an overview of the proposed *Place-Anything*. To begin, users simply need to capture a photo of a physical object or provide a brief description of a virtual one. Within minutes, our system generates a corresponding 3D mesh, accurately representing the object's form and details. Next, users upload a video, and our advanced system promptly estimates the camera's intrinsic parameters. It then reconstructs the camera poses and dense depth maps for every frame, ensuring precise alignment and integration of 3D content. At this point, users can select a region on the first frame using points or boxes, specifying where they want to place the 3D object. They can fine-tune the model's scale and orientation based on preview results, ensuring it fits naturally into the video's context. Once these steps are completed, our efficient renderer parallelly generates the corresponding multi-view images of 3D objects for each frame, blending them into the video's flow. The beauty of our system lies in its simplicity and accessibility. Unlike traditional 3D construction and rendering pipelines, all operations and interactions occur in 2D, making it easy for anyone to use. Regardless of skill level, *Place-Anything* enables users to create immersive videos effortlessly with just a few clicks. In the following, we will introduce each module.

3D Model Generation. With the given photo or text description of an object, the first step is to generate its 3D model. While existing methods ([Poole *et al.*, 2022; Raj *et al.*, 2023; Liu *et al.*, 2023]) can generate radiance fields from images or text, they suffer from limited generation speed due to volume rendering and pixel-by-pixel optimization. These fields, unlike meshes, are challenging to scale, translate, or rotate, making them unsuitable for subsequent rendering. In contrast, the advanced 3D framework [Tang *et al.*, 2023] offers easy-to-optimize 3D representations and efficient mesh extraction. We adopt it as our 3D generator, where users upload an object's description or image. We train a 3D Gaussian [Kerbl *et al.*, 2023] supervised by diffusion, then recover mesh geometry by dividing Gaussians into uniform grids, calculating weighted opacity, and using Marching Cubes for mesh extraction. Finally, we bake the colored texture map by

https://www.autodesk.com.sg/products/3ds-max/overview
https://developer.apple.com/augmented-reality/
https://developers.google.com/ar

Figure 3: Visualized examples generated by *Place-Anything*. Note that the source video of (a) is generated by SORA [Brooks *et al.*, 2024] while others are downloaded from YouTube. For privacy concerns, we covered the human faces in the realistic videos.

back-projecting rendered RGB onto the mesh surface.

Video Reconstruction. This module aims to estimate camera intrinsics, reconstruct camera poses, and generate dense depth maps for input videos. Previous SLAM systems [Campos *et al.*, 2021; Engel *et al.*, 2017], relying on key point tracking and epipolar geometry, struggle with weak or repetitive textures, making it challenging to place virtual objects on smooth surfaces. They also cannot handle videos with unknown cameras. To address these issues, we utilize [Hagemann *et al.*, 2023]. We first predict weighted optical flow between frames using RAFT [Teed and Deng, 2020]. Then a set of key frames will be filtered out based on sufficient optical flow displacement and this set of key frames will be continuously updated online every time a new frame enters in a sliding window manner. In a single window, a self-calibrating weighted bundle adjustment (SC-BA) [Hagemann *et al.*, 2023] which minimizes the reprojection errors can estimate the relative camera poses $\mathbf{G}_{ij}$, pixels depth $\mathbf{d}_i$ and intrinsic update $\Delta\mathbf{intr}$ through differentiable Gauss-Newton steps. Dense depth maps from bundle adjustment, aided by optical flow, locate any area in a video frame.

3D Target Insertion. The obtained 3D model and reconstructed video are combined to create the new video. Users can select the region for placement in the first frame via clicks or bounding boxes. Using pixel coordinates, depth values, and camera intrinsics/extrinsics, we back-project the chosen region to 3D points: $[x_i, y_i, z_i] = \mathbf{G}_i \otimes \boldsymbol{\pi}^{-1}(\mathbf{u}_i, \mathbf{z}_i, \boldsymbol{\theta})$, where $\boldsymbol{\pi}^{-1} : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}^3$ denotes the inverse projection, $\mathbf{G}_i$ denotes the camera to world projection matrix, $\mathbf{u}_i$ denotes the pixel coordinate, $\mathbf{z}_i$ is the depth value of the pixel, and $\boldsymbol{\theta}$ is the camera intrinsic. We then estimate the plane function using RANSAC on the projected 3D points. Constraints on the normal vector ensure the 3D model is placed upright. Specifically, given the normal vector as $\overrightarrow{N}$, camera position as c, and a point in the plane as p, it satisfies that $\overrightarrow{N} \cdot \overrightarrow{cp} > 0$. After rotating the 3D model to align its vertical y-axis with the normal vector, we adjust its scale based on the target area's extension range along the x- and z-axis. Finally, we render the multi-view images in parallel using pytorch3d [Ravi *et al.*, 2020] and composite the rendering results with background video.

3 Demonstrations & Results

Figure 3 presents example videos generated by *Place-Anything*, showcasing the advanced capabilities of our system. Firstly, the generated 3D model maintains strong visual coherence with the input reference images, thanks to the robust diffusion prior. Secondly, our system efficiently tracks and identifies textureless regions by leveraging optical flow to establish pixel correspondence between adjacent frames. Consequently, 3D models can be naturally integrated onto smooth surfaces like desktops or floors, regardless of the absence of corners or prominent textures. Furthermore, our model has successfully inferred the video's camera intrinsics and pose, relying solely on the harmonious and stable rendering of each frame. Lastly, mesh initialization in the third module guarantees that the inserted 3D model's scale matches the selected area perfectly.

4 Conclusion & Applications

In this paper, we present a novel system, called *Place-Anything*, which efficiently completes the entire process from 3D model production to embedding 3D models into existing videos. The simple 3D model production methods and interactive approaches make it possible for anyone to effortlessly integrate objects from their imagination or immediate environment into the creative process of any pre-existing video.

Applications. *Place-Anything* is also adaptable to various video applications, including product advertisement and marketing, video editing and post-production, and VR and AR applications. For example, brands can use *Place-Anything* to create personalized customized advertisements by inserting 3D models of their products into realistic videos. This can showcase new products in a realistic environment and can be updated at any time.

References

[Brooks *et al.*, 2024] Tim Brooks, Bill Peebles, Connor Homes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Wing Yin Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024.

[Campos *et al.*, 2021] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021.

[Engel *et al.*, 2017] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence*, 40(3):611–625, 2017.

[Hagemann *et al.*, 2023] Annika Hagemann, Moritz Knorr, and Christoph Stiller. Deep geometry-aware camera self-calibration from video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3438–3448, 2023.

[Kerbl *et al.*, 2023] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023.

[Liu *et al.*, 2023] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309, 2023.

[Poole *et al.*, 2022] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[Raj *et al.*, 2023] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023.

[Ravi *et al.*, 2020] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020.

[Tang *et al.*, 2023] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023.

[Teed and Deng, 2020] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020.