Repurposing a Relighting Network for Realistic Compositions of Captured Scenes

Baptiste Nicolet, Julien Philip, George Drettakis

Repurposing a Relighting Network for Realistic Compositions of Captured Scenes

Baptiste Nicolet
baptiste.nicolet@polytechnique.edu
Université Côte d'Azur, Inria, École polytechnique, Télécom Paris

Julien Philip
julien.philip@inria.fr
Université Côte d'Azur, Inria

George Drettakis
george.drettakis@inria.fr
Université Côte d'Azur, Inria

(a) Original Image (b) Selection (c) Composite Result

Figure 1: Example of a composition of two captured historical landmarks using our method. We extract the geometry of one (b) from a multi-view dataset, and import it into the other (a). Our method ensures coherent treatment of lighting and shadows, producing a realistic result (c).

ABSTRACT

Multi-view stereo can be used to rapidly create realistic virtual content, such as textured meshes or a geometric proxy for free-viewpoint Image-Based Rendering (IBR). These solutions greatly simplify the content creation process compared to traditional methods, but it is difficult to *modify* the content of the scene. We propose a novel approach to create scenes by composing (parts of) multiple captured scenes. The main difficulty of such compositions is that lighting conditions in each captured scene are different; to obtain a realistic composition we need to make lighting coherent. We propose a two-pass solution, by adapting a multi-view relighting network. We first match the lighting conditions of each scene separately and then synthesize shadows between scenes in a subsequent pass. We also improve the realism of the composition by estimating the change in ambient occlusion in contact areas between parts and compensate for the color balance of the different cameras used for capture. We illustrate our method with results on multiple compositions of outdoor scenes and show its application to multi-view image composition, IBR and textured mesh creation.

CCS CONCEPTS

• **Computing methodologies** → **Image manipulation**; *Image-based rendering*; Image processing.

I3D '20, May 5–7, 2020, San Francisco, CA, USA

ACM ISBN 978-1-4503-7589-4/20/05…$15.00
https://doi.org/10.1145/3384382.3384523

KEYWORDS

Deep learning, multi view, relighting, compositing

ACM Reference Format:
Baptiste Nicolet, Julien Philip, and George Drettakis. 2020. Repurposing a Relighting Network for Realistic Compositions of Captured Scenes. In *Symposium on Interactive 3D Graphics and Games (I3D '20), May 5–7, 2020, San Francisco, CA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3384382.3384523

1 INTRODUCTION

Multi-view Stereo (MVS) – together with Structure from Motion for camera calibration – is a powerful tool for capturing and visualizing real-world environments using interactive computer graphics. The resulting geometry can either be textured [Waechter et al. 2014] as an asset, or used as input for Image-Based Rendering (IBR) algorithms, allowing realistic free-viewpoint navigation [Hedman et al. 2018]. Importantly, such approaches avoid complex manual digital content creation and computationally expensive realistic rendering. While capturing such scenes is as easy as taking a few tens of photographs, it is very hard to *modify* the resulting scene. There have been some attempts to remove objects [Philip and Drettakis 2018; Thonat et al. 2016], and to change lighting [Meshry et al. 2019; Philip et al. 2019]. However, there is no obvious way to change the geometry, and in particular to realistically *combine* content from different captures into a single richer virtual environment, since this would require the lighting conditions of the different scenes to be coherent. In this paper, we present the first method that allows combination of captured scenes with coherent lighting, suitable for IBR, multi-view texturing but also image manipulation.

Naively cutting pieces from one scene to another results in several important issues. First, lighting is inconsistent between different scenes both in term of direction and intensities, since the illumination conditions are not the same in each capture. Second, regions of contact between pieces of one scene often suffer from an unrealistic look of "floating" when directly inserted into another scene. Finally, inconsistencies in camera parameters (e.g., color temperature) may occur when combining captures and negatively affect the final result.

To address lighting inconsistency, we turn to a state-of-the-art multi-view relighting method [Philip et al. 2019]. However, since it was designed for a single scene like all previous approaches, we need to adapt it to our context. Specifically, if we insert part of potentially several *source* (or *part*) scenes into a *target* (or *reference*) scene, we first need to align the lighting conditions of the source scenes to that of the target, then remove shadows from the illumination condition of the source scene. We can then synthesize new realistic shadows in the composite scene with the lighting conditions of the target scene, correctly handling overlap between shadows in the source and target scenes. Relighting, and in particular shadow removal in previous methods only works for a *single* scene; we develop a two-pass approach that enables good shadow removal and consistent shadow and lighting in the composite scene, by adapting the network of [Philip et al. 2019] so it can be used without retraining. We also provide solutions to increase the realism of contact geometry by creating more realistic contact shadows using Ambient Occlusion and allow user control of color temperature inconsistencies.

In summary, our contributions are:

- An efficient method to generalize single-scene multi-view relighting to a multi-scene setting.
- An interactive method for creating compositions of IBR scenes with coherent treatment of lighting and shadows.

We present results of our method on various compositions of outdoor scenes, and demonstrate a significant improvement over naive compositions for multi-view image editing, IBR and textured meshes.

2 RELATED WORK

Our method is built upon a multi-view relighting network. We review the previous work in areas close to our method. We first review past research on image-based scene manipulation and then review lighting estimation for composition and image relighting techniques.

2.1 Scene Manipulation and Composition

Manipulating captured scenes is a notoriously difficult problem. The focus in this area has been mostly on removing objects, followed by *inpainting* the regions revealed by removal, while maintaining multi-view consistency [Philip and Drettakis 2018; Thonat et al. 2016]. Some methods to manipulate real world scenes have been proposed, but operate in a restricted context [Zhang et al. 2016] or rely on drastic simplifications of the scene's geometry [Huang et al. 2017]. Other solutions are limited by the computational power of the devices they use [Yue et al. 2017] to generate photorealistic images.

Deep Neural Textures [Thies et al. 2019] allow the user to copy and translate an object in a *single* multi-view dataset, but does not synthesize shadows correctly. To the best of our knowledge, there is currently no previous work that attempts to compose *several* captured scenes together with consistent lighting and shadows. Lightshop [Horn and Chen 2007] allowed compositing of light fields, while recent advances in neural rendering [Flynn et al. 2019] allow compositing of light field videos [DuVall et al. 2019] but cannot handle inconsistent lighting between scenes. Neither method is adapted to our context of wide-baseline free-viewpoint navigation.

2.2 Lighting Estimation for Composition

Our goal is to realistically composite multi-view datasets; a related field of work are methods to integrate virtual objects in images. Such methods either use information recovered from inserting specific objects into the scene [Debevec 2008], or request geometric cues from the user [Karsch et al. 2011] in order to gather geometric and lighting information missing in a single image. The survey by Kronander et al. [2015], provides an extensive review of such techniques. Recent deep learning methods can infer the lighting conditions from single images [Gardner et al. 2017; Hold-Geoffroy et al. 2019, 2017; LeGendre et al. 2019] or estimate the lighting from the appearance of a specific object [Weber et al. 2018]. As the input to our method is a multi-view dataset, we can extract more information about the physical environment of each scene, and thus generate realistic compositions by mixing captured objects from several different scenes.

2.3 Image Relighting

Successful compositing of real scenes requires the use of *relighting* techniques to achieve a consistent result. Older methods rely on acquiring the intrinsic parameters of the scene either by computing a reflectance model [Yu et al. 1999] and estimated geometry segmentation [Loscos et al. 2000], or multiple photographs of the same viewpoint with varying lighting conditions [Eisemann and Durand 2004; Loscos et al. 1999]. Other methods aim at decomposing images in their intrinsic appearance parameters [Tappen et al. 2003] before computing a new rendering of the viewpoint, with changed illumination.

The multi-view setting provides additional information such as geometry estimation and multiple viewpoints of each surface. Previous methods have taken advantage of this setting to estimate intrinsic images [Duchêne et al. 2015; Laffont et al. 2013, 2012]. More recent methods rely on convolutional neural network architectures such as ResNet [He et al. 2016] to estimate intrinsic images [Sengupta et al. 2019], or directly generate the relit images [Meshry et al. 2019; Philip et al. 2019], thus avoiding the ambiguous and under-constrained model of intrinsic images.

For relighting, we will use the method described in [Philip et al. 2019], since it provides the best results with our data. The network has a notion of *source* and *target* lighting conditions, and a corresponding sun direction for each. The method takes as input the MVS geometry, shadows masks computed with this geometry for source and target conditions, and a set of illumination buffers (e.g., normals etc) computed on the fly. A first subnetwork refines the

shadow masks to assist shadow removal and resynthesis. The network is trained with synthetic data, which for the original method is manageable since all that is required is to export a pre-modeled complete scene, place cameras for reconstruction and for training relighting and then generate the MVS version of the geometry to perform supervised training of shadow map refinement. As discussed in Sec. 4.1, this is not the case in our setting.

3 OVERVIEW

Our method can be decomposed into three main components :

(1) An interactive application that allows the user to import, select, and move parts of captured scenes to create the desired composition.
(2) A deep-learning based solution to enforce consistent lighting and shadows in the composite scene.
(3) An environment compensation step to account for the modification of the surroundings of each part.

The overview of out method is displayed in Fig. 2.

3.1 Composition Interface

The input to our method consists of several multi-view datasets, each composed of many photographs (typically between 75 and 200) of a static real-world scene. These photographs are used to approximate the scene's geometry via Structure from Motion [Snavely et al. 2006] and Multi-View Stereo [Goesele et al. 2007; Reality 2018]. The first building block of our method is the composition interface. During this stage, we render a preview of the composition with a slight variation of the unstructured lumigraph algorithm [Buehler et al. 2001] (see 5.1), that allows the user to interactively edit their selection and move parts around until the composition is finalized.

In the rest of this paper, we will refer to the scene in which the user imports objects as *the reference scene*, and we will refer to the objects imported in the reference scene as *parts*. We refer to all scenes together as the *constituent* scenes, and the final combined scene as the *composite scene*. Finally the scenes used to extract parts are referred to as *part* or *original* scenes.

3.2 Consistent Composite Lighting

The composite scene contains the geometry from the different parts and the reference scene. We next proceed with the relighting step, to produce consistent lighting in the composite scene. We build on the multi-view relighting algorithm of Philip et al. [2019], which was designed for relighting a single scene. As a result, if applied naively, it cannot handle the multiple constituent parts and their corresponding different lighting and shadow levels. In addition, it cannot handle the different shadow interactions between the geometries coming from separate scenes if used on each constituent scene separately. To avoid the artifacts from such a naive solution, most notably for shadow removal, we proceed in two passes.

(1) An *offline* pass relighting the *entire* scene of origin of each *part* to match the lighting conditions of the *reference scene*. We do so by relighting all the input views.
(2) A second pass where we relight the composition, allowing us to generate cast shadows between parts. This can be either online in the novel view, or offline on all the input views, allowing interactive IBR for free-viewpoint navigation.

In most of the examples of this paper, we used the lighting conditions of the *reference* scene as target lighting conditions. Compositions are however not restricted to the lighting conditions of a constituent scene, and we can create compositions using any desired target sun direction (e.g., Fig, 8).

3.3 Environment Compensation

Consistent lighting and shadows in the composite scene are often not enough to achieve a satisfactory level of realism. As each part is extracted from a given environment in its scene of origin, and inserted in a new one, residual visual artifacts may remain even after relighting has been applied. We identified two factors that improve realism of compositions: *ambient occlusion* and *camera parameters*. We estimate the former in both the scene of origin and the reference scene for each part, and apply the corresponding compensation to the result of the second step. We provide the user with a per-scene color temperature slider, since we have no control over the camera parameters with which each scene is captured. The result of this compensation step is the final composition, suitable for IBR.

In Sec. 4 we present our approach in detail; we present our results in Sec. 5. We show results of our method on several compositions of multiple real-world scenes captured under varying lighting conditions, and compare with real-world ground truth and a previous method [Thies et al. 2019].

4 OUR METHOD

4.1 Naive solution

Direct compositing of different parts into the reference scene creates obvious visual artifacts due to the different lighting conditions in the constituent scenes (e.g., Fig. 3(a)). Multi-view relighting methods can be used to overcome this problem; we chose to work with the most recent and effective such method that uses deep learning [Philip et al. 2019].

Our first attempt was to apply the relighting network to the composite scene in a single pass. In our case, there are *multiple* source conditions, one for each part and one for the reference scene, while the target condition is common to all constituent scenes. To apply the method to the composite scene, we generate all *source* information (i.e., shadow maps, illumination buffers) on a per-pixel basis, according to the source condition of each original scene; i.e., source shadow map, sun direction, elevation etc. While the network was able to correctly predict the refined source shadow map in spite of the multiple sun orientations, it failed to completely remove shadows in some areas of the composition, as shown in Fig. 3.

Indeed, the network was never trained to deal with compositions of multiple scenes. One possible explanation for this failure is that the network cannot handle multiple *levels* of the shadows in the different scenes, since they can be significantly darker from one capture to another. Since the network is trained with single scenes, it may have learned to deal with a global value over the whole scene for shadows to be removed or added.

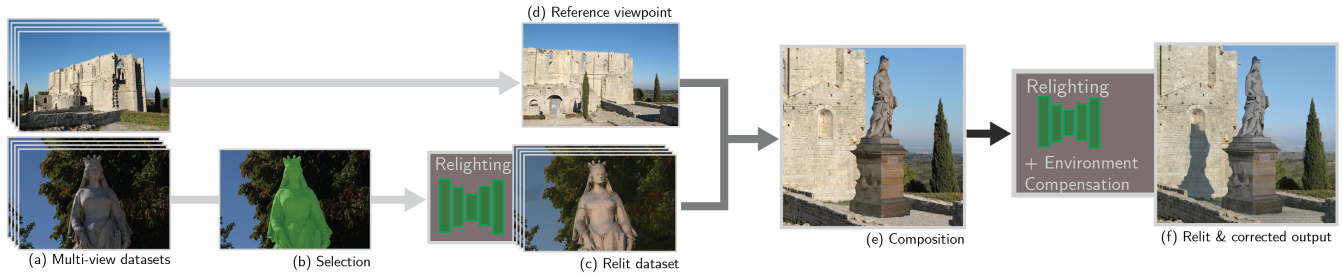


Figure 2: Overview of our method. We use multi-view datasets (a) as inputs. One dataset is considered the "reference" dataset (d), and we use its sun direction to relight the other datasets (c), then we compute a composition of the selected parts (b) into the reference scene (e). Finally, we relight the composite scene as one to synthesize shadows across scenes and we account for the changes in the environment of each scene (f)

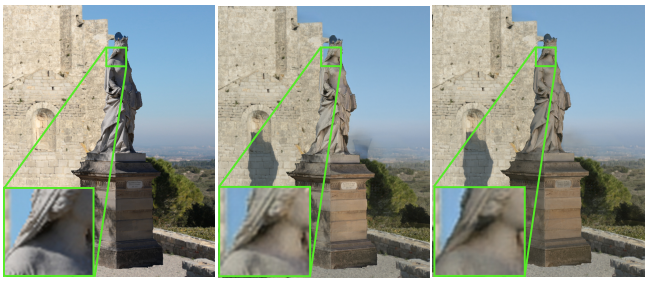


Figure 3: Example of the failure to remove shadows while directly using the relighting network of [Philip et al. 2019] in the multi-scene setting. The original shadow in (a) is not completely removed when using the network directly (b). Our approach allows us to remove it more effectively (c).

A direct solution to this issue would be to generate composite training data to re-train the network for multi-scene relighting. However, generating such data is very complex compared to the original, single-scene case. If we wanted to re-train the network, we would have to manually cut different pieces from the various training models, and create a large number of combinations of parts and references where each composition would require manual placement of pieces. This process would be even more complicated when inserting parts from several different scenes into the composite. As a result, we chose to develop a new approach that can use the original pre-trained CNN by using two separate passes, and some careful preprocessing to generate the correct illumination buffers.

4.2 Two-pass Relighting for Composite Scenes

Our two-pass solution consists of first relighting each scene individually, and then relighting the composition in a second pass. Our approach is designed to correctly remove shadows in the multi-scene setting – which cannot be handled directly by the relighting network – and to provide a consistently lit composite scene. We proceed as follows:

- Each scene is relit individually, i.e. we relight all the input images of each scene to match the lighting conditions of the reference scene. This is done once, *offline*, and requires care to only consider the selected parts of each scene. This pass generates consistent lighting for each part, but we are lacking the interactions between parts and the reference scene.
- In the second pass, we relight the current viewpoint of the composition rendered with the input images modified by the first pass. This adds cast shadows between parts, and creates fully consistent lighting and shadows.

First Pass. The goal of the first pass is to generate lighting and shadows on the selected part itself that are consistent with the reference lighting conditions (i.e., with respect to its orientation in the target scene), and in particular to correctly remove shadows of the source lighting condition of each part. We do this by adapting the relighting network of Philip et al. [2019] to relight each selected part. This pass is applied on the original part scene, but care must be taken to provide correct layers to the relighting CNN. We need to avoid unselected parts of the original scene from casting shadows onto the selected parts. We modify the shadow casting step to avoid this, see Fig. 4. Specifically, we send a shadow ray from each visible intersection point in the sun direction, to determine if the visible point is in shadow. We compute the source shadow image normally, but we compute the target shadow image only with the selected geometry, to avoid shadows cast by the non-selected geometry.

The shadow refinement part of the relighting network is thus provided with the input that will produce the desired result. At the end of this pass, we have each input image of each part scene with shadows removed, and self shadows correctly cast from the selected geometry in the reference lighting conditions (Fig. 4).

Second Pass. We can now apply the relighting network a second time on the full composite scene to cast shadows between constituent scenes, and finalize the consistent overall lighting.

This pass also requires that we carefully prepare the data sent to the relighting network. Specifically, when computing the *source* shadow images, we ray cast again only considering visible selected geometry of each part. The target shadow image is computed using the full composite scene containing all the geometry.

(a) Original Input Image (b) Relit Image - no shadow filtering (c) Relit Image - with shadow filtering

Figure 4: Illustration of our first pass relighting strategy. When relighting the input viewpoints (a) of a scene, we need to make sure that no shadow is cast by *unselected* geometry on the selected part (outlined in white in image a), resulting in hard to remove shadows (outlined in red in image b). We therefore intersect rays only against the selected geometry when computing the *target* shadow map. Since the network can not hallucinate full shadows, we end up with a "shadow-free" relit image (c).

This pass can either be done online at a given novel view, or as a preprocess on the sum of all input views (i.e., reference and part input views), and then used for IBR (see Sec. 5).

4.3 Environment Compensation

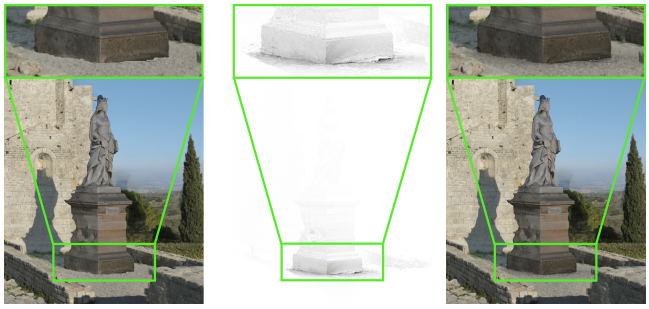
After our two pass relighting, the resulting composite has a greatly improved level of realism, for example Fig. 3(c). However there are two remaining issues.

First, parts inserted into the reference scene often appear to "float" above the ground because we have not captured the mutual shadowing effect between the two scenes in the lighting. We compensate for this problem by computing an Ambient Occlusion (AO) shift based on the geometry of the two scenes, similar in spirit to the differential rendering of Yu et al. [1999].

Second, the overall color temperature of the two scenes may be very inconsistent, and may not convey the desired visual effect. We allow the user to control the color balance of the composition to achieve the desired effect.

4.3.1 Ambient Occlusion Shift. We want to estimate the local influence of the object on its new environment. For this we use ambient occlusion. While an ambient occlusion shift is a coarse approximation of a full light transport simulation, it gives plausible results. In addition, it does not require material or explicit light information which is not available. We aim at estimating the amount of ambient occlusion to add to or remove from our composition around the newly inserted parts. We compute the original AO for each part in its scene of origin, and then compute AO in the composition. We then apply the per-pixel ratio of the two values to correct the AO of the scene. This process is illustrated in Fig. 5.

4.3.2 Color Balance. The scenes used for creating compositions may have been captured with different cameras or set up with different parameters, over which we have no control. In addition, the hue of the different parts may not convey the desired visual effect.



(a) Without AO (b) AO Shift Map (c) With AO

Figure 5: Illustration of our AO shift computation. We compute a per-pixel ambient occlusion shift (b) by casting rays in the original scene of the visible part and in the composite scene. We apply the ratio of the computed values (b) to the composite image (a), resulting in image (c).

We want to balance color on each selected part, i.e., a "layer" composed of the visible pixels of the part and black pixels everywhere else. The ambiguity between reflectance and white balance of the lighting makes it hard to adapt previous approaches that usually focus on global image statistics. We thus opted for a fast, manual correction.

We provide the user with a slider to adjust the color balance of each scene if needed. Other parameters may be adjusted in this manner, such as the exposure or gamma correction, but we found that correcting the color temperature often suffices. This modification was required for less than half of our scenes, and usually has a subtle effect, as illustrated in Fig. 6.

(a) Before Temperature Compensation (b) After Temperature Compensation

Figure 6: Illustration of our color balance compensation.

5 IMPLEMENTATION, RESULTS & LIMITATIONS

All our results are obtained on scenes reconstructed using standard Structure-from-Motion (SfM) and Multi-View Stereo (MVS) to create a geometric *proxy*. We used the commercial SfM/MVS package RealityCapture [Reality 2018] to perform reconstructions, but other solutions (e.g., colmap [Schönberger 2016]) could also be used.

5.1 Implementation Details

We use a per-pixel version of the Unstructured Lumigraph Rendering (ULR) algorithm [Buehler et al. 2001], that first renders the

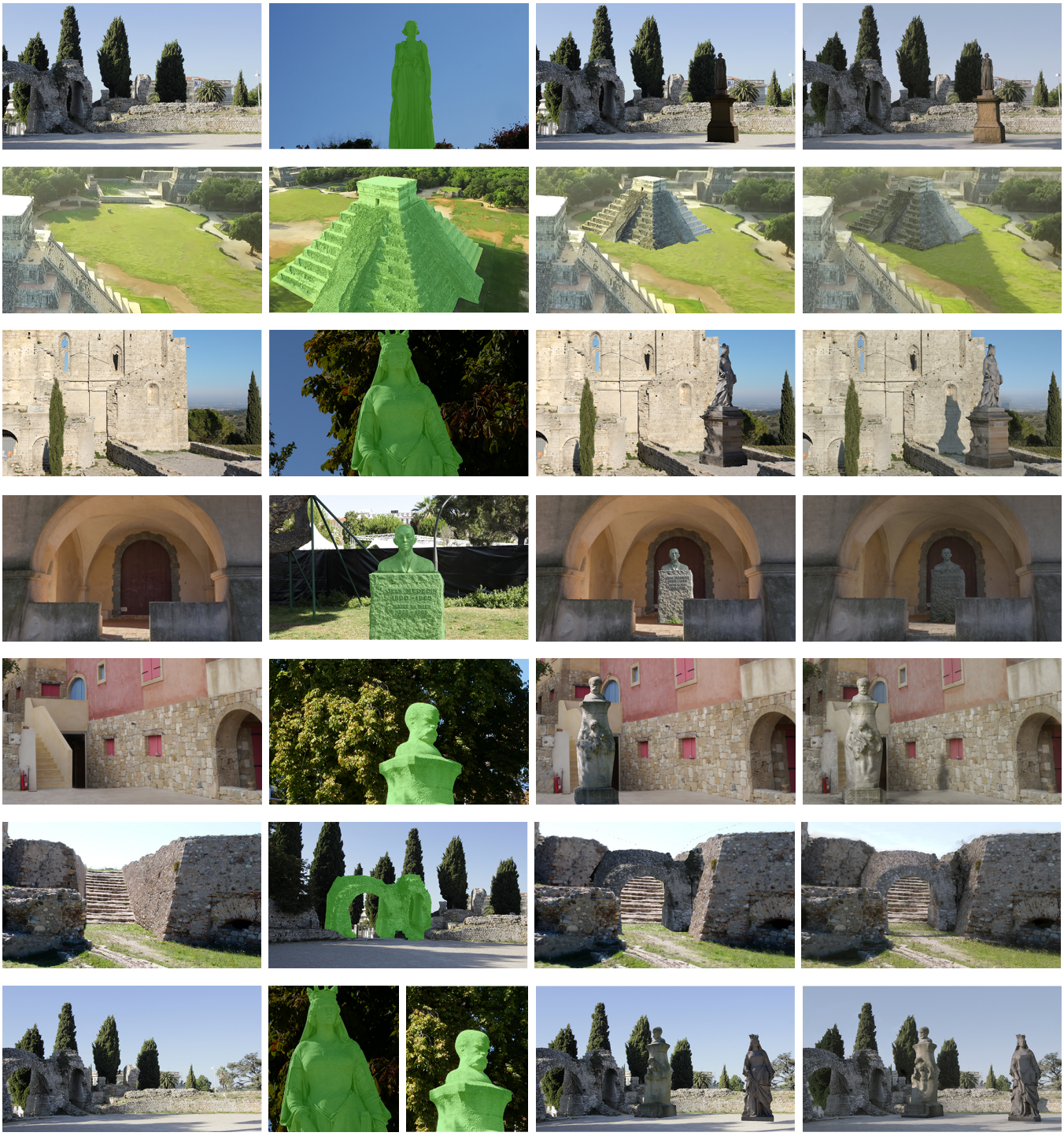


Figure 7: Examples of compositions created with our method. For each row, the leftmost image corresponds to the reference scene for our composition, the next image shows the scene from which we extract a part, highlighted in green, the next image is the naive IBR composition of the scenes, and the rightmost image is the result of the composition using our method.

proxy depth, then blends the input images per-pixel using standard ULR weights.

During interactive part placement, we render only the selected geometry of each selected part in the same render target as the reference while not clearing the depth buffer, thus allowing us to render the composition with coherent occlusions. This allows for a reasonable *real-time* preview using per-pixel ULR to create the initial version of the composite scenes; please see accompanying video. Furthermore, the user can switch to the "naive" use of the relighting network to interactively preview the cast shadows in the composite scene.

We experimented with screen-space AO, but it did not provide sufficient accuracy to compute our AO shift ratio. Instead, we use object-space AO computed by ray-casting against the reconstructed geometry.

We enable interactive exploration of the resulting composition by applying our method on all viewpoints of each constituent scene. We then use the same per-pixel ULR as for the composition preview, reprojecting modified images. In order to prevent occlusion issues when relighting a viewpoint of a given scene occluded by another part, we adjust each camera's clipping planes to be as close as possible to the selection, thus removing most of these issues, and ensuring good quality renderings when the user viewpoint is near a part's input viewpoint. We inherit the lack of coherence between relit viewpoints from Philip et al.[2019], both for temporal consistency in animation, and multi-view consistency when relighting all images of a dataset.

We show statistics of our scenes and computation times on a Intel Xeon Silver 4110 with 32GB RAM and Quadro P5000 GPU in Tab. 1. These computation times can be explained by the need to cast visibility and shadow rays through each pixel of the scene in both passes, as well as AO sample rays in the second pass, and running the result through the network. While we already use accelerating data structures and leverage SIMD instructions on the CPU side, the ray-tracing overhead could be further accelerated using ray-tracing hardware.

5.2 Applications & Results

(a) Naive Composition (b) Arbitrary Output

Figure 8: Our method can produce compositions using any provided sun direction (b).

We show three different applications: multi-view image editing, IBR and textured meshes.

The first application is multi-view image editing, where we create composite multi-view scenes. We show 7 examples of such compositions in Fig. 7, including the case of mixing 3 different scenes, and the case of a different lighting direction from the reference scene (Fig. 8).

Table 1: Computation time of some of our compositions. For each line, the first column indicates which composition we refer to (row of Fig. 7), the second is the duration of the *first pass* of our method, relighting all input viewpoints of the imported scenes. The third column (# Images) is the number of images of each scene, and the fourth is the time of the *second pass*, which allows *interactive* free-viewpoint navigation in the composition after this computation. This step is longer than the first one due to our expensive computation of ambient occlusion via ray casting, and could be accelerated (e.g., using ray-tracing hardware). The last column shows the number of input images (# Images) of the composite scene that are relit.

Scene	Pass 1	# Images	Pass 2	# Images
2nd row	7m36s	177	33m8s	354
3rd row	3m1s	75	30m56s	247
4th row	3m24s	79	23m35s	194
6th row	4m42s	85	15m26s	126

Such multi-view editing can be directly used for IBR. We can either apply the second pass relighting for each novel view on the fly (taking approx. one second per frame) or apply the pass on all the input images for all views of the reference and part scenes. When doing the latter, we can use our per-pixel ULR for free-viewpoint navigation in the composite scene. We show examples of such free-viewpoint navigation in the supplemental video. As we can see, our compositions provide a high level of realism, providing a fast way to rapidly create more complex scenes.

The last application is composite textured meshes. We first apply our method to all the input images, then re-texture the composite mesh with the relit images. In Fig. 9 we show two examples of composite textured meshes with coherent lighting.

Figure 9: Two examples of meshes textured using the relit input viewpoints.

5.3 Comparisons & Evaluation

We show comparisons with naive compositions in Fig. 7. We also show a comparison with the Deep Neural Textures approach [Thies et al. 2019], which is the only other case of composite captured scenes, albeit only with pieces of the *same* scene. As we can see Fig. 10, Deep Neural Textures do not generate cast shadows for the duplicated pieces.

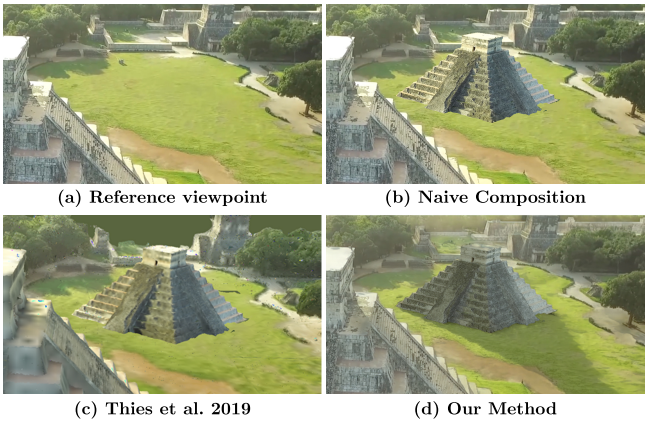


Figure 10: Comparison of our method with [Thies et al. 2019], in the case of object duplication.

Finally we provide a ground truth comparison by capturing a scene twice, once with an additional object and once without. We show our composite compared to the ground truth version in Fig. 11. While not perfect, our composition is quite close to the ground truth. Examples of remaining artifacts include small effects such as the highlight on the left arm of the statue, since the network is not designed to handle non-diffuse effects.

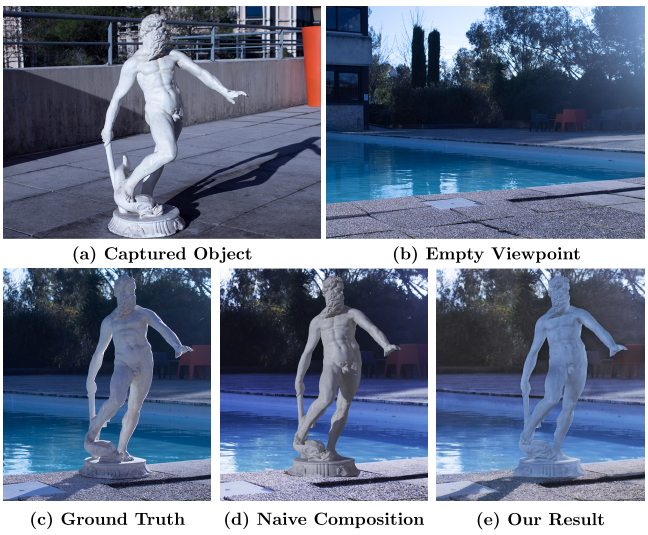


Figure 11: Real-world ground truth comparison of our method with a picture of a object inserted in a scene. While lacking some global effects, our method conveys a convincing result compared to the naive solution (d) and the initial conditions of the statue (a).

5.4 Limitations

While our method allows for fast, realistic compositions of captured scenes, it has a few limitations. First it inherits the limitations of the methods it builds on, namely Image-Based Rendering and Relighting. We have used a simple ULR-based IBR algorithm which can suffer from visual artifacts, resulting in lower quality compositions, e.g., when geometry is not well reconstructed. More sophisticated IBR algorithms, such as Deep Blending [Hedman et al. 2018] could be used, but combining or recomputing the per-view meshes for part composition is not trivial.

The geometry-aware relighting network we use is designed to relight *outdoor* scenes with cast shadows, which somewhat restricts the set of possible compositions. In addition, the network does not explicitly handle global effects. This can be seen in the real-world ground-truth comparisons (Fig. 11), where while we achieve significant improvement (b) over naive compositing (d), our method fails to account for global effects such as the reflection of light over the water's surface, which illuminates the statue from behind.

Finally, the resolution of the images plays an important role in the visual quality of the final result. Therefore, importing a part of lower resolution (e.g. captured from further away) in a higher resolution scene may impair the quality and realism of the result.

6 CONCLUSION

Interesting avenues of future work include the use of more sophisticated IBR algorithms; this involves addressing several issues such as the per-view data structures in a multi-scene context. Another interesting direction is to optimize the size of the data of the composite scene, since in many cases we can take advantage of the specific way parts have been extracted to save space.

In conclusion, we have presented a method to simply and quickly create visually compelling compositions of captured multi-view scenes, by adapting a geometry-aware relighting network to our task. To our knowledge we are the first to present such a method that can handle several scenes and provide coherent illumination in the resulting composite scene. Our solution can be used to enhance the capabilities of using IBR for free viewpoint navigation, to create more complex composite textured scenes from MVS and in some cases for photo editing.

ACKNOWLEDGMENTS

This research was funded by the ERC Advanced Grant FUNGRAPH, No 788065 (http://project.inria.fr/fungraph) The authors would like to thank S. Rodriguez for help with the IBR code base and A. Bousseau for valuable comments on a draft.

REFERENCES

Chris Buehler, Michael Bosse, Leonard McMillan, Steven Gortler, and Michael Cohen. 2001. Unstructured lumigraph rendering. In *Proceedings SIGGRAPH'01*. ACM, 425–432.

Paul Debevec. 2008. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *ACM SIGGRAPH 2008 classes*. ACM, 32.

Sylvain Duchêne, Clement Riant, Gaurav Chaurasia, Jorge Lopez-Moreno, Pierre-Yves Laffont, Stefan Popov, Adrien Bousseau, and George Drettakis. 2015. Multi-view intrinsic images of outdoors scenes with an application to relighting. *ACM Trans. on Graphics (TOG)* 34, 5 (2015).

Matthew DuVall, John Flynn, Michael Broxton, and Paul Debevec. 2019. Compositing light field video using multiplane images. In *ACM SIGGRAPH 2019 Posters*. ACM, 67.

Elmar Eisemann and Frédo Durand. 2004. Flash photography enhancement via intrinsic relighting. In *ACM transactions on graphics (TOG)*, Vol. 23. ACM, 673–678.

John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Styles Overbeck, Noah Snavely, and Richard Tucker. 2019. DeepView: High-quality view synthesis by learned gradient descent. In *Proceedings IEEE CVPR*.

Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. 2017. Learning to Predict Indoor Illumination from a Single Image. *ACM Trans. on Graphics (TOG), (SIGGRAPH Asia Conference Proceedings)* 9, 4 (2017).

Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, and Steven M Seitz. 2007. Multi-view stereo for community photo collections. In *Proceedings IEEE ICCV*. IEEE, 1–8.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings IEEE CVPR*. 770–778.

Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. 2018. Deep Blending for Free-Viewpoint Image-Based Rendering. *ACM Trans. on Graphics (TOG), (SIGGRAPH Asia Conference Proceedings)* 37, 6 (November 2018). http://www-sop.inria.fr/reves/Basilic/2018/HPPFDB18

Yannick Hold-Geoffroy, Akshaya Athawale, and Jean-François Lalonde. 2019. Deep sky modeling for single image outdoor lighting estimation. (July 2019).

Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-Francois Lalonde. 2017. Deep Outdoor Illumination Estimation. In *Proceedings IEEE CVPR*.

Daniel Reiter Horn and Billy Chen. 2007. Lightshop: interactive light field manipulation and rendering. In *Proceedings of the 2007 symposium on Interactive 3D graphics and games*. ACM, 121–128.

Jingwei Huang, Angela Dai, Leonidas J Guibas, and Matthias Nießner. 2017. 3Dlite: towards commodity 3D scanning for content creation. *ACM Trans. Graphics (TOG)* 36, 6 (2017), 203–1.

Kevin Karsch, Varsha Hedau, David Forsyth, and Derek Hoiem. 2011. Rendering synthetic objects into legacy photographs. In *ACM Trans. on Graphics (TOG)*, Vol. 30. ACM, 157.

Joel Kronander, Francesco Banterle, Andrew Gardner, Ehsan Miandji, and Jonas Unger. 2015. Photorealistic rendering of mixed reality scenes. In *Computer Graphics Forum*, Vol. 34. Wiley Online Library, 643–665.

Pierre-Yves Laffont, Adrien Bousseau, and George Drettakis. 2013. Rich intrinsic image decomposition of outdoor scenes from multiple views. *IEEE TVCG* 19, 2 (2013), 210–224.

Pierre-Yves Laffont, Adrien Bousseau, Sylvain Paris, Frédo Durand, and George Drettakis. 2012. Coherent Intrinsic Images from Photo Collections. *ACM Trans. on Graphics (SIGGRAPH Asia Conference Proceedings)* 31 (2012). http://www-sop.inria.fr/reves/Basilic/2012/LBPDD12

Chloe LeGendre, Wan-Chun Ma, Graham Fyffe, John Flynn, Laurent Charbonnel, Jay Busch, and Paul Debevec. 2019. DeepLight: Learning Illumination for Unconstrained Mobile Mixed Reality. In *Proceedings IEEE CVPR*. 5918–5928.

Céline Loscos, George Drettakis, and Luc Robert. 2000. Interactive virtual relighting of real scenes. *IEEE TVCG* 6, 4 (2000), 289–305.

Céline Loscos, Marie-Claude Frasson, George Drettakis, Bruce Walter, Xavier Granier, and Pierre Poulin. 1999. Interactive virtual relighting and remodeling of real scenes. In *Rendering Techniques' 99*. Springer, 329–340.

Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. 2019. Neural Rerendering in the Wild. In *Proceedings IEEE CVPR*. 6878–6887.

Julien Philip and George Drettakis. 2018. Plane-based multi-view inpainting for image-based rendering in large scenes. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*. ACM, 6.

Julien Philip, Michaël Gharbi, Tinghui Zhou, Alexei Efros, and George Drettakis. 2019. Multi-view Relighting using a Geometry-Aware Network. *ACM Trans. on Graphics (TOG), (SIGGRAPH Conference Proceedings)* 38, 4 (July 2019).

Capturing Reality. 2018. RealityCapture reconstruction software. https://www.capturingreality.com/Product.

Johannes L Schönberger. 2016. COLMAP: A general purpose SfM/MVS system, http://colmap.github.io.

Soumyadip Sengupta, Jinwei Gu, Kihwan Kim, Guilin Liu, David W. Jacobs, and Jan Kautz. 2019. Neural Inverse Rendering of an Indoor Scene From a Single Image. (2019).

Noah Snavely, Steven M Seitz, and Richard Szeliski. 2006. Photo tourism: exploring photo collections in 3D. In *ACM transactions on graphics (TOG)*, Vol. 25. ACM, 835–846.

Marshall F Tappen, William T Freeman, and Edward H Adelson. 2003. Recovering intrinsic images from a single image. In *Advances in neural information processing systems*. 1367–1374.

Justus Thies, Michael Zollhöfer, and Matthias Nießner. 2019. Deferred Neural Rendering: Image Synthesis using Neural Textures. *ACM Trans. on Graphics 2019 (TOG)* 38, 4 (July 2019).

Theo Thonat, Eli Shechtman, Sylvain Paris, and George Drettakis. 2016. Multi-view inpainting for image-based scene editing and rendering. In *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, 351–359.

Michael Waechter, Nils Moehrle, and Michael Goesele. 2014. Let there be color! Large-scale texturing of 3D reconstructions. In *Proceedings ECCV*. Springer, 836–850.

Henrique Weber, Donald Prévost, and Jean-François Lalonde. 2018. Learning to estimate indoor lighting from 3d objects. In *2018 International Conference on 3D Vision (3DV)*. IEEE, 199–207.

Yizhou Yu, Paul Debevec, Jitendra Malik, and Tim Hawkins. 1999. Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Siggraph*, Vol. 99. 215–224.

Ya-Ting Yue, Yong-Liang Yang, Gang Ren, and Wenping Wang. 2017. SceneCtrl: Mixed reality enhancement via efficient scene editing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. ACM, 427–436.

Edward Zhang, Michael F Cohen, and Brian Curless. 2016. Emptying, refurnishing, and relighting indoor spaces. *ACM Trans. on Graphics (TOG)* 35, 6 (2016), 174.