TransforMR: Pose-Aware Object Substitution for Composing Alternate Mixed Realities

Mohamed Kari[1,2], Tobias Grosse-Puppendahl[1], Luis Falconeri Coelho[1], Andreas Rene Fender[3], David Bethge[1], Reinhard Schütte[2], Christian Holz[3]

[1]Porsche AG, Germany
[2]University of Duisburg-Essen, Germany
[3]Department of Computer Science, ETH Zürich, Switzerland

Figure 1: TransforMR is a video see-through mixed reality system for handheld devices that performs 3D pose-aware object substitution to create meaningful mixed reality scenes, enabling applications such as alternate mixed realities or real-time virtual character animation in context. (a) From just the monocular color camera of a mobile deivce, (b) TransforMR performs instance segmentation, (c) object removal, and (d) 3D pose estimation to substitute objects with pose awareness (top right).

Abstract

Despite the advances in machine perception, semantic scene understanding is still a limiting factor in mixed reality scene composition. In this paper, we present TransforMR, a video see-through mixed reality system for mobile devices that performs 3D-pose-aware object substitution to create meaningful mixed reality scenes. In real-time and for previously unseen and unprepared real-world environments, TransforMR composes mixed reality scenes so that virtual objects assume behavioral and environment-contextual properties of replaced real-world objects. This yields meaningful, coherent, and human-interpretable scenes, not yet demonstrated by today's augmentation techniques. TransforMR creates these experiences through our novel pose-aware object substitution method building on different 3D object pose estimators, instance segmentation, video inpainting, and pose-aware object rendering. TransforMR is designed for use in the real-world, supporting the substitution of humans and vehicles in everyday scenes, and runs on mobile devices using just their monocular RGB camera feed as input. We evaluated TransforMR with eight participants in an uncontrolled city environment employing different transformation themes. Applications of TransforMR include real-time character animation analogous to motion capturing in professional film making, however without the need for preparation of either the scene or the actor, as well as narrative-driven experiences that allow users to explore fictional parallel universes in mixed reality. We make all of our source code and assets available[1].

Index Terms: Human-centered computing—Mixed / augmented reality—;—

1 Introduction

Continuous advances in *geometric* scene understanding have contributed to the *physical* coherence of virtual objects in mixed reality scenes, for example through improvements in mesh reconstruction [63], occlusion shading [7], visual-inertial odometry [20, 57], or light source estimation [59]. Research on these topics is increasingly dedicated to extracting semantic information from the real-world scene [10, 24] to enable novel mixed reality (MR) experiences [35] or context-aware interactions between virtual-world characters and the real-world environment [52].

However, both *semantic* scene understanding and *functional*—rather than physical—reasoning [67] remain hard problems. Creating an alternate reality in MR from scratch that augments a real-world city scene is a considerable challenge. Take the example of SciFi-like hover cars that pace down the streets: The mixed reality system first needs to perform scene understanding, including

[1]TransforMR code release: `https://github.com/MohamedKari/transformr`

recognizing lanes and the driving direction. To make virtual hover cars halt at a crossing zone while virtual robot pedestrians leave a real-world store entry, cross in front of the hover cars, and wait at a bus station, the system would then need to detect the crossing zones, store entries, bus stations, and sidewalks. To create such novel mixed reality experiences from scratch requires a level of scene understanding that draws on significant advances in machine perception, such as spatial scene decomposition and conceptual reasoning.

In this paper, we propose *TransforMR*, a mixed reality system for theme-guided scene transformation through *pose-aware object substitution*. TransforMR is capable of creating such meaningful mixed-reality scenes as in the example, showing and letting the user interact inside alternate mixed realities that are situated in the real-world context. TransforMR accomplishes this by repurposing existing physical objects in the scene as proxy objects that transfer their semantics in their respective environment to virtual objects. In the scenes created by TransforMR, users may attribute behavioral and environment-contextual properties of replaced real-world objects to the virtual objects. This creates semantically consistent and more plausible interactions compared to virtually augmented objects that do not inherit real-world object context and merely co-exist in the real-world surroundings.

Our system transforms visual recordings on-the-fly and is independent of a specific environment, therefore also applicable in previously unseen scenes and locations. TransforMR processes the feed of a monocular RGB camera to derive a virtual scene through a pipeline of perception, transformation, and construction. In the *perception* step, we integrate deep learning models that run on a multi-GPU-accelerated back-end, and therefore offload all processing from the mobile device. Our back-end system analyzes the streamed-in video through semantic 2D instance segmentation [12] as well as 3D human keypoint [34, 44] and 6 degrees-of-freedom vehicle pose [29] estimation. For *transformation*, TransforMR logically maps recognized objects to virtual objects according to a selected theme. In the *construction* step, TransforMR first removes the physical objects using 2D segmentation information and real-time video inpainting [23]. Lastly, TransforMR derives the final scene from projecting the theme-specific objects into the scene using the 3D pose information. Our use of inpainting allows transformed objects to occupy less display space than the removed objects, reconstructing the background where needed.

Figure 1 shows TransforMR in action. Here, a user is exploring the transformation of reality by looking through the tablet while freely walking through the real world. TransforMR substituted all pedestrians and vehicles from the scene with semantically corresponding objects from the "Animals" theme. As depicted in the figure, the transformed objects are shown in the context of all physical surroundings, allowing the user to maintain their frame of reference for safe navigation (e.g., when climbing stairs).

To the best of our knowledge, TransforMR is the first system with the capability of 3D pose-aware object substitution in unbounded, unprepared, and unseen environments with visually complex scenes. We enable this unprecedented live mixed reality experience with the sole requirement of a single RGB camera, making our system suitable for broad applicability in lower-end phones or tablets and high-end devices alike.

Contributions

Taken together, we make the following contributions in this paper:

- Pose-aware object substitution as a novel technique to creating meaningful, theme-based alternate mixed reality scenes with virtual objects that assume behavioral and environment-contextual properties of replaced real-world objects,
- A camera-to-display system architecture, implementation, and design rationale for "TransforMR", a distributed mixed reality system that adapts, unifies, and integrates a series of deep learning-based 2D and 3D scene perception architectures as well as video inpainting for operating in-the-wild in real-time in unseen environments on commodity mobile devices,
- A parallization architecture employing three-step pipeline parallelism and three-fold task parallelism to achieve near-real-time operation of the integrated computer vision models at approximately 15 frames per second,
- An evaluation and discussion of the qualitative and technical aspects of TransforMR,
- Applications of TransforMR that comprise real-time character animation in real-world context and narrative-driven, consumptive experiences of alternate mixed reality scenes.

2 Related Work

TransforMR composes the virtual scene with the context provided by the real scene. Previous work that uses real-world scene context for virtual-scene composition includes geometry- and depth-aware AR, superposition-based AR, and physicality-aware VR. As pose-aware object substitution features an object removal procedure, we consider diminished reality as well as as a pipeline of diminished and augmented reality as related areas. Given the transformative character of TransforMR, we consider 3D scene reconstruction and transformation, as well as visual transformation as related research.

2.1 Context-Aware Mixed Reality

Geometry-aware AR and *Depth-aware AR* as implemented in Apple's ARKit[2] and Google's ARCore[3] enable applications to additively render new objects into a real scene while respecting its the geometrical context, providing capabilities for collision detection between real and virtual objects (e.g., virtual rain drops hitting the real ground, or virtual balls bouncing off the real walls), occlusion shading (i.e., partially covering virtual objects by real objects), or depth-of-field effects (e.g., bokeh or fog), using depth-from-motion approaches or depth sensors [7, 56]. Moreover, Nuernberger et al. [42] explore a concept for aligning virtual objects with edges in the real world. However, semantically meaningful augmentations are still difficult to achieve, in particular in an automated fashion without user guidance such as anchor setting, since they require the system to not only have an accurate geometric representation, but also a purposeful semantic representation beforehand. Furthermore, they do not feature object replacement or removal procedures.

Superposition-based AR is a widely established approach for anchoring a virtual object with a concealed real-world object, that is seen for the first time (e.g., a face) or has been incorporated into a set of reference objects. Examples include Annexing Reality by Hettiarachchi and Wigdor [14], Snapchat Lenses[4] and Apple Animojis[5]. This, of course, leads to the restriction that rendered objects must fully cover the real objects by being of similar shape and larger size. Especially when replacing multiple objects in a scene that are close to each other, this enlargement constraint cannot be satisfied without unnatural overlapping effects. In MediaPipe, Lugaresi et al. [30] extract object poses from shoes and chairs and superimpose virtual objects based on the pose information.

Physicality-Aware VR is concerned with enabling a virtual-reality experience that allows roaming the virtual environment while avoiding physical obstacles through redirected walking. Yang et al. [62] present DreamWalker, a system that - given a real-world destination – guides the user through a pre-authored virtual environment while

[2] ARKit:https://developer.apple.com/documentation/arkit

[3] ARCore:https://developers.google.com/ar/discover

[4] https://www.engadget.com/2020-02-20-snapchat-ground-lenses-floor-is-lava.html

[5] https://www.apple.com/newsroom/2019/12/clips-now-features-memoji-and-animoji-new-stickers-and-more/

avoiding physical static and moving obstacles. Cheng et al. [5] present VRoamer, a system that procedurally generates VR environments on-the-fly, constrained by the perceived physical obstacles. While the aforementioned work focuses on avoiding the interaction mismatches between the real and virtual environment with respect to walking, the subfield of tangible AR deals with reducing interaction mismatches with respect to touching, e.g., using physical proxy objects [38, 48, 50].

2.2 Dimished Reality

Diminished Reality aims at removing objects from a scene [11, 13, 19, 21, 32, 36, 40]. However, none of these systems simultaneously satisfy the three imposed constraints of (1) dimishing with only a real-time stream of monocular RGB information, (2) dimishing moving objects, and (3) dimishing all instances of a certain object class. More importantly however, diminished reality is not at all concerned with the simultaneous estimation of 3D poses or rendering replacement objects instead.

Piping Diminished Reality and Augmented Reality While our notion of Pose-Aware Object Substitution requires both diminishing real objects from and placing virtual objects in the scene, a naïve pipeline of separately applying a Diminished Reality system and then applying a geometry-aware AR system is fundamentally insufficient to achieve the envisioned result for TransforMR. Specifically, such a pipeline would not be able to achieve semantical coherence between the virtual objects and the environment. This inability results from the lack of semantic information or 3D pose information in either of the systems. For example, such a DR/AR pipeline could not create scenes in which vehicle-like objects move along real driving lanes, because the concept of a "lane" is not known to the AR system.

2.3 3D Scene Reconstruction and Transformation

Litany et al. [28] present an approach for semantics-invariant scene transformation based on point clouds in rooms. Izadinia et al. [18] present a system for transforming a single RGB image of a furnished room into a corresponding composition of CAD models, drawing on a database of such models. Their system is based on multiple applications of convolutional networks for object detection, scene segmentation into "ceiling", "right wall", "middle wall", etc. to derive room geometry, and estimating the objects' feature vectors for similarity measurements against the database. Finally, they apply a render-and-match approach to refine 3D poses. Avetisyan et al. [1] pursue the same objective, however rely on joint layout and object estimation. Shapira and Freedman [49] present Reality Skins, a system for generating virtual environments based on a 3D scan of a room. These setups are incompatible with our goal of allowing untethered open-world on-the-fly applications.

2.4 Visual Transformation

Non-photorealistic 2D rendering ranges from traditional convolutions to neural video style transfer [4, 17, 47] to create cartoon, night vision, art, or similar effects. However, by design, these approaches generally modify texture without the possibility to perform transformations such as replacing vehicles with animals.

Video-to-video translation has been used for input-conditioned creation of photorealistic videos. Thies et al. [54] present Face2Face for real-time facial reenactment. Wang et al. [58] present Few-Shot vid2vid for facial or body reenactment or converting semantic maps or human pose models to image sequences.

2.5 Summary

While we have identified *technically similar* or *conceptually similar* work in the preceding paragraphs, we argue that all of it is *functionally different* in that it does not aim at providing a real-time, on-the-fly user experience through transforming a real scene into a semantically transformed, yet isomorphic scene, which preserves the correspondence between real and virtual objects. These functional differences technically manifest in fundamentally different system architectures that for example do not comprise components for semantic mapping, object replacement or removal using temporal and spatial information, have less computation needs and do not need investigate offloading to multiple backend GPUs, nor deal with pipeline parallelism.

3 TransforMR: Design & Architecture

In this paper, we propose a novel method for composing meaningful mixed reality scenes by transforming a real-world scene into an alternate reality through *pose-aware object substitution*. Figure 2 shows our proposed substitution procedure comprising object detection and pose estimation, object removal, object mapping, and pose-aware object rendering. In the following, we consider our design objectives, describe our system architecture for pose-aware object substitution, and describe its technical implementation.

3.1 Design Objectives

TransforMR builds on recent advances in computer vision to realize pose-aware object substitution under a set of design objectives that enable in-the-wild application:

- *Environment Independence*. We want TransforMR to operate on previously unseen scenes without prior preparation of the environment. This means our system cannot rely on on-site-installed camera systems known from room-scale experiences.
- *Handheld Display Rendering*. To allow users to comprehend the correspondences between the virtual and real objects, we display transformed scenes on a handheld display. This assorts well with the objective of environment independence, as handheld displays are, generally speaking, more broadly applicable in public spaces than head-mounted displays.
- *Mobile Device Compatibility*. As we envision broad applicability of TransforMR by enabling users to employ their own mobile device, we impose the constraint of compatibility with common smartphones or devices without the need for additional on-person hardware. As a consequence, perception must rely on a monocular RGB camera only and does not include time-of-flight sensor information.
- *Real-Time Execution Ability*. Being restricted to one monocular RGB camera only entails the requirement of compute-intensive machine vision methods for object and pose detection, and object removal. This conflicts with the limited hardware capacities present in mobile devices. Nonetheless, we subject our system to real-time execution, that is we abstain from a post-capture AR approach and instead aim at processing frames in a real-time fashion.

3.2 Pose-Aware Object Substitution Architecture

Figure 3 gives an overview of our pose-aware object subsitution system. The overall system input is given by a real-time sequence of monocular RGB frames representing observations from the real environment. The overall system output is a sequence of RGB frames showing the transformed scene. As illustrated in Fig. 4, TransforMR performs a series of perception, transformation, and construction operations as described in the following.

3.2.1 Perception: 3D Pose Estimation

We intend to render virtual objects into the conceived scene with the same pose as the physical object being replaced. Therefore, it is necessary to estimate the 3D poses of those physical objects first. While certain problems such as instance segmentation and 2D bounding box detection can be solved with general-purpose models

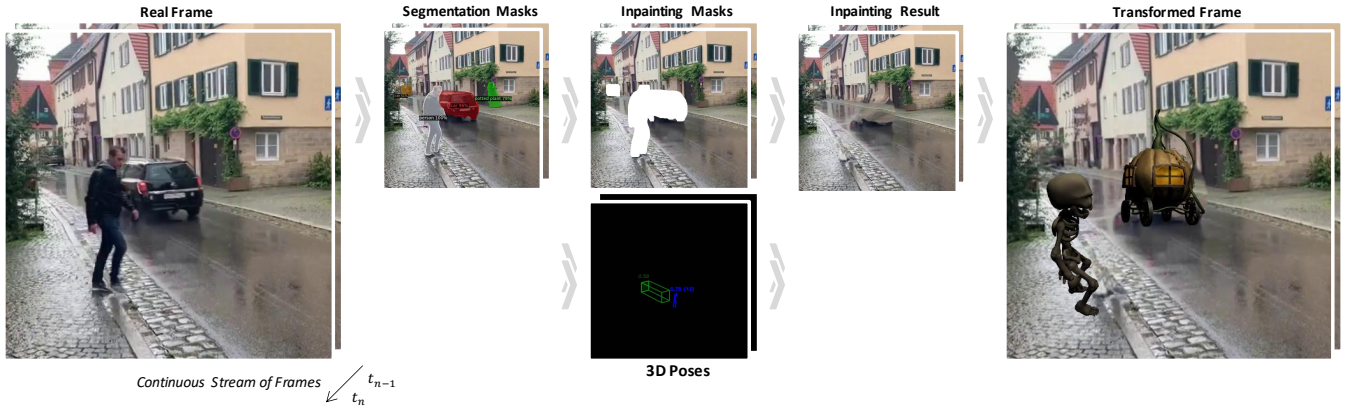

Figure 2: An example of continuously creating a Halloween-themed alternate reality through pose-aware object substitution, based on object removal through deep-learning-based instance segmentation and video inpainting, and as well as 3D pose estimations for humans and vehicles. Please also refer to the video figure.

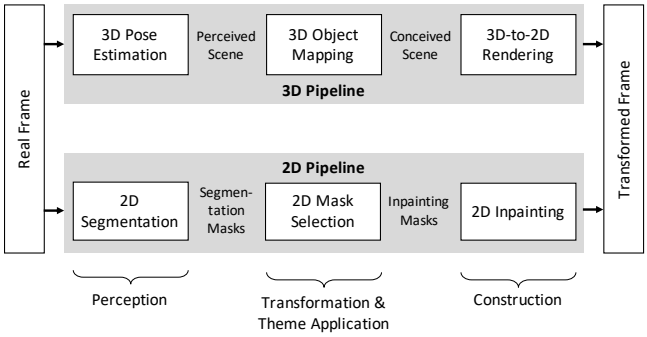

Figure 3: Component diagram of the TransforMR system implementation. TransforMR runs 2D and 3D pipelines in parallel with both pipelines performing perception, transformation, and construction steps. The 2D pipeline comprises instance segmentation and non-look-ahead video inpainting in image space. The 3D pipeline estimates object poses in 3D camera space, and renders objects at the same position with the same pose. Mapping is guided by themes through class-specific mapping instructions.

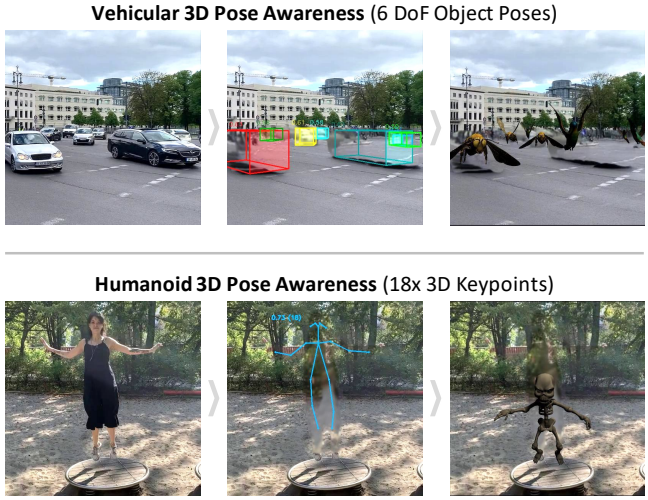

Figure 4: Examples for our 3D-pose-aware object substitution approach. 3D pose estimation is performed for vehicles (top) and humans (bottom). Vehicles poses are estimated as oriented 3D bounding boxes. Human poses are estimated as 3D joint keypoints.

trained on high-diversity datasets, 3D pose estimation algorithms are predominantly designed for specific purposes.

For detecting 3D poses – more specifically, 6 degrees-of-freedom pose – of vehicles in traffic scenes [16, 25, 29, 37], we employ SMOKE (Single-Stage Monocular 3D Object Detection via Keypoint Estimation) by Liu et al. [29]. Relying on a CenterNet-like network architecture with deep layer aggregation and deformable convolutions [6, 64, 68] for feature extraction, SMOKE directly regresses location and orientation parameters from a single monocular RGB frame without an intermediary step of inferring 2D object proposals first. Since SMOKE operates on single frames and ignores the temporal dimension, we employ a distance-based tracking filter on the estimated centroid in 3D space to infer cross-temporal object identity.

For detecting 3D poses – more specifically, 18 different keypoints in 3D space – of humans [33, 34, 45, 46, 61], we employ *Lightweight Human Pose Estimation* by [45] which is based on a previously presented architecture [34], however, modified in order to decrease inference duration by using a MobileNet-like [15] feature extractor. It is noted that Android's ARCore doesn't support human pose estimation and Apple's ARKit supports 3D human pose estimation just for a single person and on recent chipsets [55] only. Lightweight Human Pose Estimation can estimate the poses of multiple persons simultaneously. As with the 3D vehicle pose estimation, a distance-based tracking is applied to infer object identity across a sequence of frames. Attached to the feature extractor are 2D and 3D keypoint detection stages.

Both pose estimation models are encapsulated as independent modules with the same abstract interface of consuming a single frame and thereupon returning a list of pose detections. State from processing previous frames is managed internally by each module. While models should predict pose information accurately relative to the camera, we adjust for different relative scales across the models by maintaining a model-specific scaling factor. Figure 4 visualizes the 3D awareness in image space, achieved by the pose estimation procedures described.

3.2.2 Perception: 2D Segmentation

For the object removal procedure, we rely on hole inpainting. We determine the holes to be inpainted through instance segmentation the Mask R-CNN algorithm [12]. Each instance segmentation mask corresponds to a detected instance and represents a bitmap the size of the input image which indicates the presence or absence of a pixel belonging to the respective instance. We use the *detectron2* implementation with a ResNet-50/FPN feature extractor.

Figure 5: Overview of the specified SciFi, Halloween, Animals, Prehistory, and Classic Cars themes. The different themes feature 3D vehicular and humanoid object models.

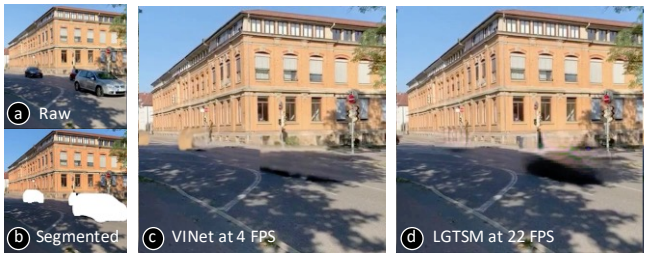

Figure 6: Exemplaric comparison of the two alternative real-time video inpainting methods, integrated in TransforMR. (a) Based on the monocular RGB frame, (b) TransforMR runs 2D instance segmentation to produce the inpainting bitmap. (c) With our adaptions in VINet, inpainting can operate at approximately 4 FPS without lag at a single VINet model instance and approximately 7 FPS with two load-balanced VINet model instances. VINet yields visually coherent inpainting for large masks. (d) With our adaptions in LGTSM, a chunk size of 4 frames, and downsampling to a width and height of 200 px, can operate at approximately 22 FPS, however at the cost of visual coherence for larger holes.

3.2.3 Transformation: Theme-Guided Semantic Mapping

Transformed scenes are a function of the object-reduced input frame, the current object *detections* estimated by the system, and the *theme* selected by the user. A detection comprises estimations of the class, 3D pose information, and possibly additional information of a real-world object. The theme comprises *class-specific mapping instructions*. Each mapping instruction is scoped to a detection class supported by the perception module. It indicates which virtual object models can be rendered in lieu of the real object. A single class, e.g., *car*, can be mapped to different virtual object models, thus producing diverse transformations. In order to retain the mapping between a physical object in a frame and its previous mapping in preceding frames, a substitution state storing tracking IDs of detected objects and the corresponding virtual instances is managed. Figure 5 shows the themes we have prepared for use in TransforMR.

With these themes, users can employ TransforMR to either create their own narratives or to interactively consume provided narratives, e.g., created by a narrative provider in a certain context such as a museum or zoo. We discuss these narratives in Section 5.

3.2.4 Construction: Video-Inpainting-based Object Removal

We accomplish the goal of object removal through real-time video inpainting where the inpainting mask in each frame is filled by estimating the globally and locally most plausible pixel values. Inpainting masks are derived from the instance segmentation bitmaps estimated as described above.

Since classical methods of image inpainting generally yield implausible results for larger holes or lead to inconsistent or flickering inpainting across consecutive video frames, we turn to learning-based video-inpainting methods [3, 8, 22, 23, 26, 39, 43, 51, 60, 65, 66]. Generally relying on optical-flow estimation to reconstruct the path of a pixel value through the temporal dimension, information is propagated from previous or future frames into the region to be filled. Filtering out methods which, by design, expect knowledge of all frames in advance, or methods with uncompetitive frame rates that are therefore unfit for our real-time objective, we integrated VINet [22, 23] and inpainting based on Learnable Gated Temporal Shift Modules (LGTSM) [3] as alternative methods into our system architecture.

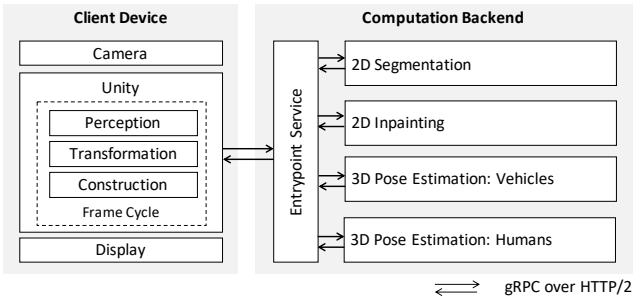

Figure 7: Deployment diagram of TransforMR. RGB frames are captured on the client-device camera, shipped over the network to a GPU-accelerated host that performs computation-intensive operations and returns pose estimations for all relevant objects as well as the inpainted frames. Rendering of the virtual 3D objects takes place on the client device.

While VINet is originally designed to peek five frames into the future, we adapted the inference logic so that no look-ahead is performed, thus reducing system lag. Further, while LGTSM-based inpainting originally chunks up a video into smaller batches, and operates on these smaller batches, we adapted the inference logic to feed-back inpainted frames into the next input chunk with an all-intact inpainting mask, thus yielding a frame-by-frame inpainting method suitable for real-time application.

VINet has an inference latency of approximately 250 ms, however produces visually coherent results for larger holes. Using a load-balancing approach and distributing frames across two VINet model instance allows to operate inpainting at 7 FPS. With our adaptions for frame-by-frame real-time inference, LGTSM-based inpainting has an inference latency of only approximately 45 ms. However, while VINet can propagate information of long gone frames to the current frame inpainting through state in the recurrent LSTM units, LGTSM-based inpainting only convolutes on the last three frames for inpainting and fills the remaining information generatively. This results in less coherent results for larger holes. Both frame rates refer to exclusive usage of an NVIDIA Tesla V100 GPU.

3.2.5 Construction: 3D Rendering

With the inpainted frame, the detections including the 3D pose information, and the selected theme as an input, we run the 3D scene rendering in the Unity graphics engine to obtain the transformed frame.

3.3 Technical Implementation

As all computer-vision models in TransforMR employ convolutional neural networks are therefore computationally intensive, we run them on four NVIDIA Tesla V100 GPUs using CUDA. Each model runs in a separate Docker container on the cloud server.

We implement a central access point to the TransforMR backend, also running in a container that distributes a single request across the different models and integrates their responses in the perception result that is sent back to the client. Client-server and inter-container communication is implemented using gRPC[6]. While the perception is offloaded to the cloud, the transformation and the construction

[6] gRPC: `https://github.com/grpc/grpc`

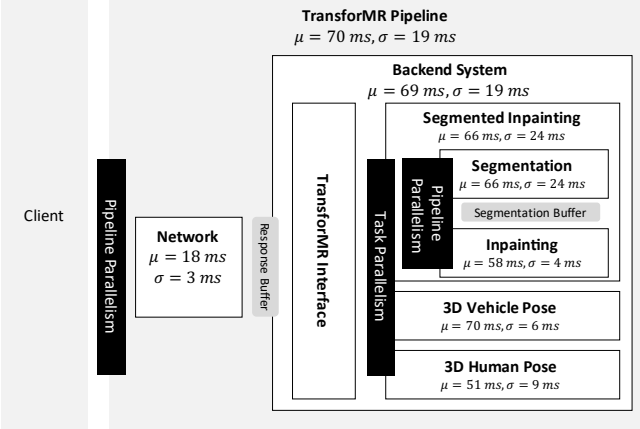

Figure 8: Breakdown of the network and inference latencies in TransforMR. Note, that latencies of components in a parallelized pipeline do not sum up, but are given by the maximum of the individual pipeline component latencies. Network RTT was measured from our institutional local area network in Zürich to an AWS EC2 VM in Frankfurt. It was determined in a network test setup by immediately returning the received frame. Benchmarking was performed on the scene that is shown in Figure 6.

module are running on the terminal client in Unity[7]. This setup allows shifting computationally demanding load from the device into the cloud, thus relaxing the hardware and software requirements on individual users' local devices. Also, the architecture enabled us to quickly test different state-of-the-art algorithms which were not optimized for mobile devices. The downside of this approach is that going over the network adds a lag, depending on the network conditions. Figure 7 exhibits a deployment diagram.

Using LGTSM-based inpainting instead of VINet, our computation backend achieves a processing frame rate of approximately 15 FPS and a system lag of 3 frames. We achieve this by a multi-faceted parallelization architecture. *First*, we employ pipeline parallelism between the network and the computation backend, so that we send the next RGB frame while the previous frame is still being processed by the backend. *Second*, we employ threefold task parallelism for a) 3D pose estimation of vehicles, b) 3D pose estimation of humans, and c) the segmentation and inpainting pipeline. As inpainting depends on segmentation, we cannot run segmentation and inpainting in parallel for a single given frame, but we can run instance segmentation, while the previous frame is still being inpainted. That is, *third*, we employ pipeline parallelism between the instance segmentation and the inpainting. In summary, at each frame cycle, the backend runs inference through four neural networks in parallel. We restrict frame buffer size at the central entrypoint service to size 1, so that the backend throttles the client automatically if frames are served faster than they are processed.

4 Preliminary Evaluation

4.1 Participants

We conducted a preliminary user evaluation with 8 participants ($n = 8$, $n_{female} = 2$, ages 21 to 55, $\mu_{age} = 28.8$, $\sigma_{age} = 10.9$), 5 of which from our institution. No one of the participants was involved in the project. Two participants had limited experience with AR applications, another two had extensive experience. The participants received a small gratuity after the evaluation sessions.

[7]Unity: `https://unity.com/`

4.2 Procedure

Participants were first shown a video of a scene that was transformed using TransforMR with the SciFi theme. They were informed that the system is capable of transforming humans and vehicles. We employed quality-optimized inpainting. 5 out of the 8 evaluation sessions were conducted at a busy street near our institution, the other three were conducted at a traffic-calmed street with a bordering park farther away. On-site, participants were instructed to employ an Apple iPad Pro, 12.9 inches, to explore the surroundings at their discretion, switching through all 5 provided themes. Overall, each evaluation session took 14 to 25 minutes. Afterward, participants were to fill out the Augmented Reality Immersion questionnaire by Georgiou and Kyza [9], extended by TransforMR-specific questions, by stating agreement on a 7-point Likert scale. In addition, we conducted a semi-structured interview to gain insight into their subjective impressions.

4.3 Results and Discussion

Figure 9 exhibits a subset of the evaluation items. All participants found the experience enjoyable or very enjoyable.

Transformation Themes. Three participants (P1, P6, P7) liked the animal theme most with all of them stating they liked the wing-flapping animation of the bee and the hummingbird. Two participants (P3, P4) expressed that they liked the Halloween theme best, one of them (P3) stating that it suited the atmosphere of the tree-lined road very well. From this, we derive the idea of *context-specific themes* as part of a *consumptive interaction pattern*. E. g., visitors of an amusement park might want to alter their environment by substituting other visitors with suitable cartoon characters. We elaborate on this in the applications subsec. 5.2.

Interaction. Most interestingly, three participants using the system asked if were okay for other participants to join in on the activity as "actors" in order to explore how humans were transformed. From this, we derive the notion of the *director-actor interaction pattern* for *creating narratives*, described in subsec. 5.1. Furthermore, P5 described it as "awkward to direct the tablet on random people unknown to him". He noted that this "awkwardness" was reduced when people would approach him or when multiple people were in the scene. Extending the perception range to other objects, e.g., animals, could allow users to focus on other alterations beyond humans. P6 noted that he would have liked to see objects from very up close, but getting too close would make them vanish. Whether the 3D object pose can be inferred from an object depicted in macro-perspective depends on the model. For example, the 3D vehicle pose estimation model we employ requires the vehicles 3D center to be in the frame. On the other hand, the 3D human pose estimation model can also infer the pose, for example, from the head of a human only.

Alteration Experience. Three (P2, P4, P6) out of the 8 participants stated that they would have liked to see augmentation effects known from classic AR apps too, while the other 5 participants (P1, P3, P5, P7, P8) stated they prefer the app to only show virtual objects that have a real correspondence. One participant (P8) who indicated the preference of no augmentations said, that it felt "more real to known that what [she] saw was really there, in a way, and not just imagined". One participant (P2) liked classic augmentations in addition to the real-world object alterations stated that this would be "interesting for situations when there are no cars or humans around". However, with 5 users abstaining from the wish to add Augmented Reality objects to the scene, we conclude that upholding the correspondence awareness between virtual and real objects exhibits a particularly exciting alteration experience. P7 reported in a highly positive mood that he felt as if he was "in an apocalypse movie with mutated bees". However, P6 expressed disconcertment on the fact that the prehistoric theme transforms not only parked cars, but also even moving vehicles to wooden carriages without a draft animal. From this, we see that one has to distinguish between *spatial plausibility* and

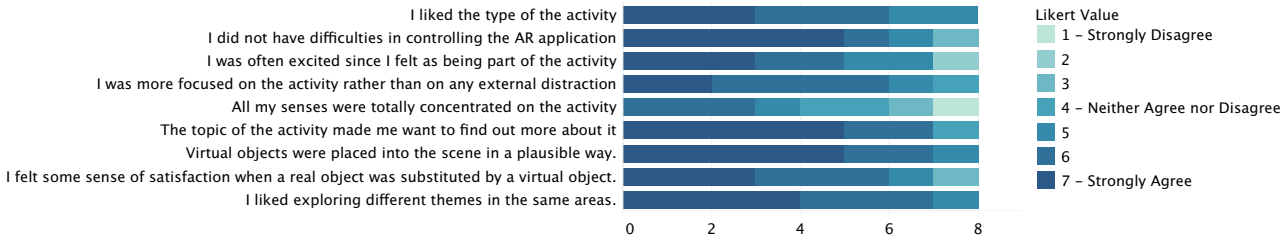

Figure 9: Likert ratings of the participants on statements from the ARI questionnaire (upper 6 statements) by Georgiou and Kyza [9] and on TransforMR-specific statements (lower 3 statements)

Figure 10: Different scenes transformed towards different themes. TransforMR enables users to roam previously unseen, unbounded, unprepared, and changing environments featuring multiple humans and vehicles, all at the same time, transformed by a user-chosen theme.

semantic plausibility. The above-mentioned statement on the bee shows that semantic implausibility can be a source of additional excitement or disconcertment. Since users can take agency by selecting certain themes and decide on the camera direction, they can influence the plausibility level achieved.

Alteration Consistency. While all participants considered the alteration consistency positive overall, two participants (P3, P1) noted that sometimes the human objects "were off and jumping around" resp. "switching identities". Incremental improvement of the prediction models could help stabilize prediction. Two participants (P2, P4) said that they noticed significant differences in the object removal quality. P2 recounted that multiple times he "didn't even notice that there was actually a person crossing right in front of them" when at other times it seemed as if "it just blurred the object". While inpainting still remains a difficult problem in computer-vision research, we believe it could be interesting to also add a post-capture AR experience, as known from recent releases in Google ARCore[8]. Here, instead of harder real-time inpainting, this would allow using slower, but better inpainting operating *on all frames* including future ones, thus enriching the optical flow information.

[8]`https://developers.google.com/ar/develop/java/recording-and-playback/introduction`

Focus. While all participants stated that they primarily focused on the display, 4 out 8 participants (P1, P5, P6, P8) recalled they would regularly check the real scene and compare it with the transformed scene and the other 4 participants (P2, P3, P4, P7) would compare it to the real scene sometimes. P1 and P2 recounted that they would look up especially to search for new situations. These findings cement our conclusion that users see Correspondence Awareness as an important part of the experience.

5 Applications

In the above evaluation, we ascertained that users like to employ TransforMR for real-time in-context character animation with multiple users as well as for the exploration of an alternate mixed reality around them. From this, we derive the usage patterns of *consuming narratives* and *creating narratives*.

5.1 Creating Narratives

As seen in Figure 11, users can follow a *director-actor interaction pattern*, where one user takes the role of the director and one or more other users take the role of actors. The directing user will then employ their smartphone to capture the acting users in context, thus creating a transformed scene. In so doing, users can collaboratively

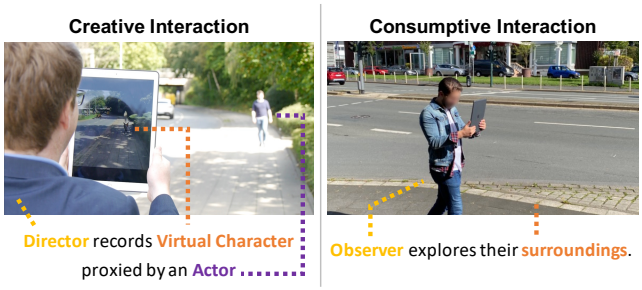

Figure 11: **Left**: Following a director-actor interaction pattern, users can perform real-time character animation, reminiscent of motion capturing in professional film making studios, however in real-world context instead, in order to create their own narratives. **Right**: Users can also explore and transform their surroundings single-handedly. In this pattern, interaction is given through theme selection, environment navigation, and camera direction.

tell stories about virtual characters, offered in the theme, who walk through parks or school buildings, ride the bus, or do grocery shopping. This kind of role play is only achievable through the concept of pose-equivalent character substitution, allowing interesting new scenes composed of interactions between characters such as anthropomorphic animals, robots, celebrities, or avatars, proxied by real humans.

5.2 Consuming Narratives

Consuming predefined narratives with *context-specific themes*, in particular location as an important context property, could, for example, enable users to experience time-travel through the history of traffic in a certain region e.g., by visualizing evolving eras of urban traffic through-out history. Similarly, visitors at a historic site could transform their environment towards a theme replacing other visitors with models of humans that are in line with the historic culture in question. Or possibly, visitors of an art museum might want to experience the museum alongside the original artists instead of their real fellow visitors. By previously authoring such themes with the corresponding 3D object models, maintainers of a facility could enable visitors to immerse deeper into their stay.

6 Limitations

Increasing Prediction Accuracy. Object and pose prediction models as well as inpainting are at the core of the TransforMR system. The plausibility of the transformed scenes is therefore inherently limited by the models' prediction performances. The model for human pose estimation is limited to humans no farther away than approximately 15 meters and unreliable when it comes to correctly detecting keypoints of cyclists. Tracking of objects is also challenging, in particular in if they are located very near to each other. Figure 12a) and b) show examples. Improvements could be achieved, e.g., by adding temporal recurrency to the detection models [27], using a learning-based object tracker [16], employing detection models with physical constraints [46], or building on advances in correspondence estimation [53].

Improving Visual Coherence. Modern augmented reality frameworks feature techniques that can improve the visual coherence of the generated imagery, in particular to create realistic illuminations and occlusions [31]. Since shadows are purposefully not removed in the presented pipeline, virtual objects "reuse" the shadow cast by the original objects, thus alleviating part of the illumination problem [2]. However, as seen in Figure 12b), occlusions are not detected in TransforMR. Instead, heavily occluded objects are likely to not be detected by the 3D detection models, thus leaving the object untransformed. Adding light source estimation for more realistic

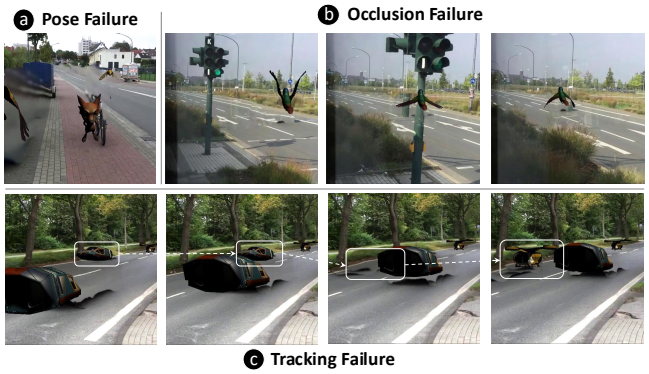

Figure 12: Perceptually challenging situations can cause visually incoherent transformed scenes. (a) Intricate poses in humans, e.g., of cyclists, can cause incoherent renderings of the virtual avatars. (b) Heavily occluded objects are likely to not be detected, thus yielding scenes that are mostly coherent with the depth in the scene. However, slightly occluded real objects might be detected anyways, leading to renderings of virtual objects that are negligent of occlusions. (c) Objects that move considerably across a couple of frames, possibly even hidden behind other objects, can cause a discontinuation in the instance tracking chain.

illumination and in particular adding a depth map estimation for occlusion shading [56] could improve the geometric plausibility of transformed scenes.

Targeting Head-Mounted Displays. We have designed TransforMR as a mobile AR system so to ensure that users can always keep track of correspondences between virtual and real objects. However, future work might explore the opportunities to employ the pipeline for use in head-mounted displays (HMD). While we hypothesize that such an approach might be beneficial in terms of full immersiveness, it remains an open research question how to minimize latency of such a system generally, and inference latency of all the deep-learning models specifically, so that the higher frame rate demand required for safe and smooth HMD experiences can be met.

Estimating World Coordinates. TransforMR's perception pipeline only estimates the location and pose of objects with respect to the camera, but does not track their location in world coordinates. Therefore, the system does not differentiate between ego motion and object motion. Using SLAM approaches [41], either visually or by means of the devices inertial sensors, could help the system to estimate world-relative object movement. This would allow movement-aware substitutions, e. g. with virtual horses that either stand, walk, trot, or career, depending on the velocity of the real object.

Enriching Themes and Narratives. TransforMR has been implemented with support for detecting and estimating the 3D pose of vehicles and humans. Adding pose estimation for other indoor or outdoor object classes such as animals, trees, chairs, etc. will enable richer experiences. Furthermore, we see a potential for complementing pose-aware object substitution with pose-aware plane substitutions, e.g., to transform streets into water, lava, or lunar soil. Furthermore, we see a high potential for increasing immersiveness through Audio AR features that emphasize narratives for the consumptive usage of TransforMR.

7 Conclusion

In this paper, we have presented TransforMR, a system that performs pose-aware object substitutions to create meaningful alternate mixed reality scenes. Designed under the objectives of environment independence, correspondence traceability, mobile device compatibility, and real-time execution, we proposed a cloud-assisted architecture

comprising computer-vision models for 2D instance segmentation, 3D pose estimation for vehicles, 3D pose estimation for humans, speed or quality-optimized alternatives for object removal through inpainting, as well as comprising a semantic mapping procedure and the 3D rendering.

In a preliminary user evaluation, we found that users particularly like to employ TransforMR for real-time character animation, reminiscent of motion capturing in professional filmmaking, however operating without preparation of either the scene or the actor.

While gaming-oriented Mixed Reality applications of today mainly borrow *3D geometry* from the real world and register objects based on planes and visual features, TransforMR heads towards mixed reality experiences that do not only incorporate geometrical but also semantical information from the real-scene context into the composition of the mixed-in virtual scene. In the future, we expect to see more research that explores concepts to realize semantics-driven mixed reality, complementary to or derived from this work on reusing semantical object embeddings.

One major hurdle in realizing semantics-driven mixed reality scenes lies in the computational demand. More specifically, to extract possibly multiple layers of semantics from camera input using demanding neural networks, a single, mobile-device GPU can present an insurmountable bottleneck. In TransforMR, we showcased the integration of four neural networks without being subject to mobile resource limitations. We believe that future research can draw on the feasibility of such a cloud mixed reality approach. To enable the community to reproduce the system and to build on our work, we make all of our source code and assets available on GitHub.

REFERENCES

[1] A. Avetisyan, T. Khanova, C. Choy, D. Dash, A. Dai, and M. Nießner. Scenecad: Predicting object alignments and layouts in rgb-d scans. *arXiv preprint arXiv:2003.12622*, 2020.

[2] L. Casas, M. Fauconneau, M. Kosek, K. Mclister, and K. Mitchell. Image based proximate shadow retargeting. In *Proceedings of the Conference on Computer Graphics & Visual Computing*, CGVC '18, p. 43–50. Eurographics Association, Goslar, DEU, 2018. doi: 10.2312/cgvc.20181206

[3] Y.-L. Chang, Z. Y. Liu, K.-Y. Lee, and W. Hsu. Learnable gated temporal shift module for deep video inpainting. In *Proceedings of the 30th British Machine Vision Conference*, 2019.

[4] D. Chen, J. Liao, L. Yuan, N. Yu, and G. Hua. Coherent online video style transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1105–1114, 2017. doi: 10.1109/ICCV.2017.126

[5] L. Cheng, E. Ofek, C. Holz, and A. D. Wilson. Vroamer: Generating on-the-fly vr experiences while walking inside large, unknown real-world building environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 359–366, 2019. doi: 10.1109/VR.2019.8798074

[6] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 764–773, 2017. doi: 10.1109/ICCV.2017.89

[7] R. Du, E. Turner, M. Dzitsiuk, L. Prasso, I. Duarte, J. Dourgarian, J. Afonso, J. Pascoal, J. Gladstone, N. Cruces, et al. Depthlab: Real-time 3d interaction with depth maps for mobile augmented reality. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pp. 829–843, 2020. doi: 10.1145/3379337.3415881

[8] C. Gao, A. Saraf, J.-B. Huang, and J. Kopf. Flow-edge guided video completion. In *Proc. European Conference on Computer Vision (ECCV)*, 2020.

[9] Y. Georgiou and E. A. Kyza. The development and validation of the ari questionnaire: An instrument for measuring immersion in location-based augmented reality settings. *International Journal of Human-Computer Studies*, 98:24–37, 2017. doi: 10.1016/j.ijhcs.2016.09.014

[10] J. Grubert, T. Langlotz, S. Zollmann, and H. Regenbrecht. Towards pervasive augmented reality: Context-awareness in augmented reality. *IEEE transactions on visualization and computer graphics*, 23(6):1706–1724, 2016. doi: 10.1109/TVCG.2016.2543720

[11] J. Guida and M. Sra. Augmented Reality World Editor. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST*, 2020. doi: 10.1145/3385956.3422125

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969, 2017. doi: 10.1109/ICCV.2017.322

[13] J. Herling and W. Broll. Advanced self-contained object removal for realizing real-time diminished reality in unconstrained environments. In *2010 IEEE International Symposium on Mixed and Augmented Reality*, pp. 207–212, 2010. doi: 10.1109/ISMAR. 2010.5643572

[14] A. Hettiarachchi and D. Wigdor. Annexing reality: Enabling opportunistic use of everyday objects as tangible proxies in augmented reality. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, p. 1957–1967. Association for Computing Machinery, New York, NY, USA, 2016. doi: 10.1145/2858036.2858134

[15] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. 2017.

[16] H.-N. Hu, Q.-Z. Cai, D. Wang, J. Lin, M. Sun, P. Krahenbuhl, T. Darrell, and F. Yu. Joint monocular 3D vehicle detection and tracking. In *Proceedings of the IEEE international conference on computer vision*, pp. 5390–5399, 2019. doi: 10.1109/ICCV.2019.00549

[17] H. Huang, H. Wang, W. Luo, L. Ma, W. Jiang, X. Zhu, Z. Li, and W. Liu. Real-time neural style transfer for videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 783–791, 2017. doi: 10.1109/CVPR.2017.745.

[18] H. Izadinia, Q. Shan, and S. M. Seitz. Im2cad. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5134–5143, 2017. doi: 10.1109/CVPR.2017.260

[19] N. Kawai, T. Sato, and N. Yokoya. Diminished reality based on image inpainting considering background geometry. *IEEE Transactions on Visualization and Computer Graphics*, 22(3):1236–1247, 2016. doi: 10.1109/TVCG.2015.2462368

[20] C. Kerl, J. Sturm, and D. Cremers. Dense visual slam for rgb-d cameras. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2100–2106. IEEE, 2013. doi: 10.1109/IROS.2013.6696650

[21] D. Kido, T. Fukuda, and N. Yabuki. Diminished reality system with real-time object detection using deep learning for onsite landscape simulation during redevelopment. *Environmental Modelling and Software*, 131(June):104759, 2020. doi: 10.1016/j.envsoft.2020.104759

[22] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon. Recurrent temporal aggregation framework for deep video inpainting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(5):1038–1052, 2019. doi: 10.1109/TPAMI.2019.2958083

[23] D. Kim, S. Woo, J.-Y. Lee, and I. So Kweon. Deep video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5792–5801, 2019. doi: 10.1109/CVPR.2019.00594

[24] K. Kim, M. Billinghurst, G. Bruder, H. B.-L. Duh, and G. F. Welch. Revisiting trends in augmented reality research: A review of the 2nd decade of ismar (2008–2017). *IEEE transactions on visualization and computer graphics*, 24(11):2947–2962, 2018. doi: 10.1109/TVCG.2018.2868591

[25] J. Ku, A. D. Pon, and S. L. Waslander. Monocular 3D object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11867–11876, 2019. doi: 10.1109/CVPR.2019.01214

[26] S. Lee, S. W. Oh, D. Won, and S. J. Kim. Copy-and-paste networks for deep video inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4413–4421, 2019. doi: 10.1109/ICCV.2019.00451

[27] M. Lin, L. Lin, X. Liang, K. Wang, and H. Cheng. Recurrent 3D pose sequence machines. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 810–819, 2017. doi: 10.1109/CVPR.2017.588

[28] O. Litany, T. Remez, D. Freedman, L. Shapira, A. M. Bronstein, and R. Gal. Asist: Automatic semantically invariant scene transformation. *Comput. Vis. Image Underst.*, 157:284–299, 2017. doi: 10.1016/j.cviu.2016.08.002

[29] Z. Liu, Z. Wu, and R. Tóth. Smoke: Single-stage monocular 3D object detection via keypoint estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 996–997, 2020. doi: 10.1109/CVPRW50498.2020.00506

[30] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann. Mediapipe: A framework for building perception pipelines, 2019.

[31] D. Mandl, K. M. Yi, P. Mohr, P. M. Roth, P. Fua, V. Lepetit, D. Schmalstieg, and D. Kalkofen. Learning lightprobes for mixed reality illumination. In *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 82–89, 2017. doi: 10.1109/ISMAR.2017.25

[32] S. Meerits and H. Saito. Real-time diminished reality for dynamic scenes. In *2015 IEEE International Symposium on Mixed and Augmented Reality Workshops*, pp. 53–59, 2015. doi: 10.1109/ISMARW.2015.19

[33] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, M. Elgharib, P. Fua, H.-P. Seidel, H. Rhodin, G. Pons-Moll, and C. Theobalt. Xnect: Real-time multi-person 3D motion capture with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 39(4):82–1, 2020. doi: 10.1145/3386569.3392410

[34] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, S. Sridhar, G. Pons-Moll, and C. Theobalt. Single-shot multi-person 3D pose estimation from monocular rgb. In *2018 International Conference on 3D Vision (3DV)*, pp. 120–130. IEEE, 2018. doi: 10.1109/3DV.2018.00024

[35] O. Miksik, V. Vineet, M. Lidegaard, R. Prasaath, M. Nießner, S. Golodetz, S. L. Hicks, P. Pérez, S. Izadi, and P. H. Torr. The semantic paintbrush: Interactive 3D mapping and recognition in large outdoor spaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3317–3326, 2015. doi: 10.1145/2702123.2702222

[36] S. Mori, S. Ikeda, and H. Saito. A survey of diminished reality: Techniques for visually concealing, eliminating, and seeing through real objects. *IPSJ Transactions on Computer Vision and Applications*, 9, 06 2017. doi: 10.1186/s41074-017-0028-1

[37] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka. 3D bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7074–7082, 2017. doi: 10.1109/CVPR.2017.597

[38] T. Muender, A. V. Reinschluessel, S. Drewes, D. Wenig, T. Döring, and R. Malaka. Does it feel real? using tangibles with different fidelities to build and explore scenes in virtual reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300903

[39] R. Murase, Y. Zhang, and T. Okatani. Video-rate video inpainting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1553–1561. IEEE, 2019. doi: 10.1109/WACV.2019.00170

[40] Y. Nakajima, S. Mori, and H. Saito. Semantic Object Selection and Detection for Diminished Reality Based on SLAM with Viewpoint Class. *Adjunct Proceedings of the 2017 IEEE International Symposium on Mixed and Augmented Reality, ISMAR-Adjunct 2017*, pp. 338–343, 2017. doi: 10.1109/ISMAR-Adjunct.2017.98

[41] Y. Nakajima, S. Mori, and H. Saito. Semantic object selection and detection for diminished reality based on slam with viewpoint class. In *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 338–343. IEEE, 2017. doi: 10.1109/ISMAR-Adjunct.2017.98

[42] B. Nuernberger, E. Ofek, H. Benko, and A. D. Wilson. Snaptoreality: Aligning augmented reality to the real world. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, p. 1233–1244. Association for Computing Machinery, New York, NY, USA, 2016. doi: 10.1145/2858036.2858250

[43] S. W. Oh, S. Lee, J.-Y. Lee, and S. J. Kim. Onion-peel networks for deep video completion. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4403–4412, 2019. doi: 10.1109/ICCV.2019.00450

[44] D. Osokin. Real-time 2D multi-person pose estimation on cpu: Lightweight openpose, 2018. doi: 10.5220/0007555407440748

[45] D. Osokin and M. Ageeva. Real-time 3D multi-person pose estimation demo. `https://github.com/Daniil-Osokin/lightweight-human-pose-estimation-3d-demo.pytorch`, 2019.

[46] D. Rempe, L. J. Guibas, A. Hertzmann, B. Russell, R. Villegas, and J. Yang. Contact and human dynamics from monocular video. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[47] M. Ruder, A. Dosovitskiy, and T. Brox. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, pp. 26–36. Springer, 2016. doi: 10.1007/978-3-319-45886-1_3

[48] P. Schulz, D. Alexandrovsky, F. Putze, R. Malaka, and J. Schöning. The role of physical props in vr climbing environments. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, p. 1–13. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300413

[49] L. Shapira and D. Freedman. Reality skins: Creating immersive and tactile virtual environments. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 115–124, 2016. doi: 10.1109/ISMAR.2016.23

[50] A. L. Simeone, E. Velloso, and H. Gellersen. Substitutional reality: Using the physical environment to design virtual reality experiences. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3307–3316, 2015. doi: 10.1145/2702123.2702389

[51] R. Szeto, M. El-Khamy, J. Lee, and J. J. Corso. Hypercon: Image-to-video model transfer for video-to-video translation tasks. *arXiv preprint arXiv:1912.04950*, 2019.

[52] T. Tahara, T. Seno, G. Narita, and T. Ishikawa. Retargetable AR: Context-Aware Augmented Reality in Indoor Scenes based on 3D Scene Graph. *Adjunct Proceedings of the 2020 IEEE International Symposium on Mixed and Augmented Reality, ISMAR-Adjunct 2020*, pp. 249–255, 2020. doi: 10.1109/ISMAR-Adjunct51615.2020.00072

[53] F. Tan, D. Tang, M. Dou, K. Guo, R. Pandey, C. Keskin, R. Du, D. Sun, S. Bouaziz, S. Fanello, P. Tan, and Y. Zhang. Humangps: Geodesic preserving feature for dense human correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1820–1830, June 2021.

[54] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. doi: 10.1109/CVPR.2016.262

[55] Unity. Unity docs - about ar foundation. `https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.0/manual/index.html`, 2019.

[56] J. Valentin, A. Kowdle, J. T. Barron, N. Wadhwa, M. Dzitsiuk, M. Schoenberg, V. Verma, A. Csaszar, E. Turner, I. Dryanovski, J. Afonso, J. Pascoal, K. Tsotsos, M. Leung, M. Schmidt, O. Guleryuz, S. Khamis, V. Tankovitch, S. Fanello, S. Izadi, and C. Rhemann. Depth from motion for smartphone ar. *ACM Trans. Graph.*, 37(6), Dec. 2018. doi: 10.1145/3272127.3275041

[57] J. Wald, K. Tateno, J. Sturm, N. Navab, and F. Tombari. Real-time fully incremental scene understanding on mobile platforms. *IEEE Robotics and Automation Letters*, 3(4):3402–3409, 2018. doi: 10.1109/LRA.2018.2852782

[58] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, J. Kautz, and B. Catanzaro. Few-shot video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[59] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger. Elasticfusion: Real-time dense slam and light source estimation. *The International Journal of Robotics Research*, 35(14):1697–1716, 2016. doi: 10.1177/0278364916669237

[60] R. Xu, X. Li, B. Zhou, and C. C. Loy. Deep flow-guided video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3723–3732, 2019. doi: 10.1109/CVPR.2019.00384

[61] Y. Xu, S. Zhu, and T. Tung. Denserac: Joint 3D pose and shape

estimation by dense render-and-compare. pp. 7759–7769, 10 2019. doi: 10.1109/ICCV.2019.00785

[62] J. Yang, C. Holz, E. Ofek, and A. D. Wilson. Dreamwalker: Substituting real-world walking experiences with a virtual reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, pp. 1093–1107, 2019. doi: 10.1145/3332165.3347875

[63] X. Yang, L. Zhou, H. Jiang, Z. Tang, Y. Wang, H. Bao, and G. Zhang. Mobile3drecon: Real-time monocular 3D reconstruction on a mobile phone. *IEEE Transactions on Visualization and Computer Graphics*, 26(12):3446–3456, 2020. doi: 10.1109/TVCG.2020.3023634

[64] F. Yu, D. Wang, E. Shelhamer, and T. Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2403–2412, 2018. doi: 10.1109/CVPR.2018.00255

[65] H. Zhang, L. Mai, N. Xu, Z. Wang, J. Collomosse, and H. Jin. An internal learning approach to video inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2720–2729, 2019. doi: 10.1109/ICCV.2019.00281

[66] R. Zhang, W. Li, P. Wang, C. Guan, J. Fang, Y. Song, J. Yu, B. Chen, W. Xu, and R. Yang. Autoremover: Automatic object removal for autonomous driving videos. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(7), 2020. doi: 10.1609/aaai.v34i07.6982

[67] Y. Zhao. *A Quest for Visual Commonsense: Scene Understanding by Functional and Physical Reasoning*. University of California, Los Angeles, 2015.

[68] X. Zhou, D. Wang, and P. Krähenbühl. Objects as points. 2019.