FlyAR: Augmented Reality Supported Micro Aerial Vehicle Navigation

Stefanie Zollmann, Christof Hoppe, Tobias Langlotz, and Gerhard Reitmayr

Fig. 1. Augmented Reality supported flight management for aerial reconstruction. (Left) Aerial reconstruction of a building. (Middle) The depth estimation for a hovering MAV in the distance is complicated due to missing depth cues. (Right) Augmented Reality provides additional graphical cues for understanding the position of the vehicle.

Abstract—Micro aerial vehicles equipped with high-resolution cameras can be used to create aerial reconstructions of an area of interest. In that context automatic flight path planning and autonomous flying is often applied but so far cannot fully replace the human in the loop, supervising the flight on-site to assure that there are no collisions with obstacles. Unfortunately, this workflow yields several issues, such as the need to mentally transfer the aerial vehicle's position between 2D map positions and the physical environment, and the complicated depth perception of objects flying in the distance. Augmented Reality can address these issues by bringing the flight planning process on-site and visualizing the spatial relationship between the planned or current positions of the vehicle and the physical environment. In this paper, we present Augmented Reality supported navigation and flight planning of micro aerial vehicles by augmenting the user's view with relevant information for flight planning and live feedback for flight supervision. Furthermore, we introduce additional depth hints supporting the user in understanding the spatial relationship of virtual waypoints in the physical world and investigate the effect of these visualization techniques on the spatial understanding.

Index Terms—Augmented reality, micro aerial vehicles, visualization

✦

1 INTRODUCTION

Micro aerial vehicles (MAVs) such as quad- or octocopters are an emerging technology. While small commodity devices are designed to perform simple movements in the near-field controlled by a simple remote control, professional devices such as octocopters equipped with automatic balancing technology, professional GPS and inertial sensors focus on mid- and far-distance applications. These devices are built to even transport small additional payload such as a camera. There are several application areas that can benefit from professional MAVs such as collecting a set of aerial views for reconstructing an area of interest (Figure 1, Left). The resulting 3D information can be used for various industrial applications, for instance for construction site monitoring [29]. To obtain high reconstruction quality within a limited flight time, automatic flight path planning methods help to record a high number of images from meaningful viewpoints [13].

However, automatic methods for flight path planning usually only plan ideal viewpoints and send them as a list of waypoints to the MAV. They do not consider how the MAV exactly moves from one waypoint to the next nor take obstacles into account. In order to address this problem, a lot of research on autonomous flying has been done. For instance, vision-based autonomous mapping and exploration methods allow autonomous navigation in indoor and outdoor environments [7, 6]. Nevertheless, these methods for autonomous flying are still a field of active research and are so far not used for professional industrial purposes, in particular not in urban areas. Even for research tests, usually the autonomous flight sessions are supervised. This means so far these techniques have not the ability to fully replace the human in the loop supervising a flight session on-site for avoiding collisions with physical obstacles in the worst case, for instance if the MAV looses network connection or computation fails. In addition, in some countries autonomous flying is only allowed with inter-visibility. For that reason, professional MAVs come with a remote control that allows the supervisor to interfere in an emergency.

Supplementary, 2D map interfaces on mobile devices allow for prior inspection of flight paths and live flight supervision showing the current position and next waypoints. But in that case, the user still has to establish the spatial relationship between the positions on the 2D map and the physical environment. With such a workflow it can be challenging to avoid obstacles since the user has to either mentally map the physical obstacles to the 2D map or vice versa in order to transfer the flight path from the map to the physical environment. In particular, this is an issue in situations where the 2D map does not re-

---

- *Stefanie Zollmann is with Graz University of Technology. E-mail: zollmann@icg.tugraz.at.*
- *Christof Hoppe is with Graz University of Technology, E-mail: hoppe@icg.tugraz.at.*
- *Tobias Langlotz is with Graz University of Technology and with University of Otago, E-mail: langlotz@icg.tugraz.at.*
- *Gerhard Reitmayr is with Graz University of Technology. E-mail: reitmayr@icg.tugraz.at.*

Manuscript received 12 September 2013; accepted 10 January 2014; posted online 29 March 2014; mailed on 1 May 2014.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

flect the current situation on-site, as it is often the case for dynamic construction environments. Furthermore, it is difficult to understand the distance of the MAV, if it is too far away and depth cues are not available (Figure 1, Middle) as well as to identify planned waypoints that are located behind an obstacle and therefore not longer in the field of view - a situation that supervisors highly try to avoid for security reasons.

Using Augmented Reality (AR) as an interface for supporting the navigation of aerial vehicles has the advantage of retaining the spatial relationship between flight relevant data and the physical environment by overlaying the waypoints in real-time onto a camera image representing the physical world (Figure 1, Right). Obstacles on the path are visible in the camera image and if depth information is available conflicts can be highlighted. The benefit of AR for such navigation tasks was already identified by Kasahara et al. when navigating a robot by physically moving an AR interface in relation to the robot [17]. This approach was demonstrated to work well in near range to the user. Unfortunately, this approach cannot be applied in this way for mid and far range navigation, since (1) the MAV's positions are often outside the reaching range of the user, and (2) providing depth estimation for flying objects at these distances is difficult.

In this paper, we address the aforementioned issues by introducing FlyAR, an AR interface for inspecting and creating flight paths of a MAV, as well as for live flight supervision. We discuss which data provided by the MAV is suitable for AR visualization and how to access and visualize this data (Section 3). In order to support the spatial understanding and recognition of critical waypoints, we investigated how physical depth cues, such as occlusion, can be integrated into the visualization (Section 3.2). In addition, we discuss the difficulties that appear when observing flying objects in the distance [3] and propose a set of additional graphical hints that help to improve the depth perception of the waypoints (Section 3.3). By integrating the data access, the visualization techniques and 6DOF model-based tracking into one AR system (Section 4), we are able to investigate the benefit of such a system for different applications (Section 5). Furthermore, we will present a user study showing the positive effect of the FlyAR system on the spatial understanding.

2 RELATED WORK

Additionally to the research focusing on methods for automatic flight path planning [13] and autonomous flying [7], there is a large amount of related work that investigates remotely controlling of mobile robots or MAVs. One of the earlier works aimed at reducing the workload of users in navigational tasks for MAVs by visualizing a set of flight specific information on a PDA, such as a graphical wing view that abstracts roll and altitude [21]. Ishii et al. proposed a laser gesture interface that allows a user to control a mobile robot in indoor environments with natural gestures by combining gestures and a laser pointer to give the robot instructions [15]. The gestures are recognized by an external ceiling camera and allow one to specify targets or tasks for the robot.

Crescenzio et al. presented an interface for supervising missions of a MAV [2]. The interface comprises planning tools based on a 2D map control panel and a 3D visualization on a projection wall for supporting spatial understanding. Additionally, audio feedback is used to provide live feedback. Hashimoto et al. proposed the TouchMe system that allows one to use a touch interface for controlling a remotely located mobile robot [10]. The interface presents an AR overlay consisting of the remote camera view containing the robot and virtual handles. By touching the virtual handles on the screen, the user manipulates different parts of the robot. One of findings of the TouchMe project was the need for more meaningful visualizations for understanding the current state of the robot.

In AR, meaningful and comprehensible visualization methods were investigated by several research groups. There is a broad body of work about methods supporting users in understanding spatial relationships of virtual and physical information within AR. For instance, graphical hints such as object-aligned virtual cutaways [30] or Magic Lenses [22] are used to support depth perception of users in subsurface X-Ray AR applications. Wither and Hoellerer suggested using virtual shadow planes and color-encoded markers for supporting depth perception while placing virtual annotations [28]. Wither et al. extended their work on supporting the user in understanding spatial relationships by providing a 2D map and an AR view containing the annotations at the same time [27]. In order to support depth estimation of occluded targets in an AR visualization, Livingston et al. introduced a set of additional graphical hints. These hints are based on different mapping techniques that encode the depth of a virtual target in its appearance [19]. Additionally, a virtual ground plane provides supplementary depth cues. To enhance this effect the virtual targets are anchored to the ground plane using virtual lines. Dey et al. investigated different methods for providing spatial understanding of an occluded target object in an AR view [5]. There is also some work that describes how to support users with graphical hints in manipulating a remote robot. Nawab et al. introduced visual cues for supporting the teleoperation of a remote robot [20]. They use a color-coded joystick to control the position and the orientation of the robot and described a positive effect of integrating the same color-coded coordinate system into the AR visualization of remote camera views of the robot.

Recently, Kasahara et al. proposed the concept of exTouch, an AR-based interface for controlling actuated objects, such as a flying drone [17]. Their interface enables the user to manipulate the actuated marker-tracked objects by touch gestures. Nevertheless, this interaction method only works if the robot is in proximity to the user. As soon as the device moves further away this kind of interaction become more unreliable. Markers attached to aerial vehicles were also used to visualize navigation-related information, such as force vectors [23]. A disadvantage of marker-based methods is that they will not work at medium- to far-field distances, due to the requirement that the marker has to be visible to the camera.

Other approaches focus on superimposing aerial views from the drone's perspective with target information for military applications [16, 9]. The drone in these applications usually navigates at bigger heights and far away from obstacles. Thus aerial views are suitable for these navigation tasks. However, for applications that have the goal to capture close views from urban areas, relying exclusively on aerial views can be dangerous.

To our best knowledge, to date there is no work on supporting the understanding of spatial relationships in mid and far field navigation of micro aerial vehicles. More precisely, there is no approach available that augments the direct physical environment of the user with the additional information about planned and current positions of such a vehicle. Furthermore, there is only a small amount of work that aims to support depth estimation of flying virtual objects in AR and there is no work available that investigates additional depth hints for visualizing the spatial relationship of an aerial vehicle, waypoints and the user's position within the physical world.

3 VISUAL HINTS FOR SUPPORTING FLIGHT MANAGEMENT

During a flight session, there is a lot of flight-relevant information available. We can subdivide them into two types of information: 1) current state information and 2) planned and prospective state information. Current state information provides live feedback about the aerial vehicle and is directly published by the aerial vehicle, including data such as GPS data, inertial measurement unit (IMU) data and recorded camera images. GPS data contains information about the current position of the vehicle, while IMU data messages provide information about the device's orientation. This data is periodically updated and published by the MAV. Additionally, the MAV publishes if a camera image was recorded and information about this image.

Planned and prospective state information of the aerial vehicle comprises waypoints, pathlists and camera control parameters. Waypoints and pathlists describe the planned geo-referenced positions of the aerial vehicle as well as the planned yaw at this position. For the aerial vehicle that we use (Asctec Falcon 8[1]) yaw is the only angle that can be controlled, since the device is supposed to have a fixed roll and

---

[1] http://www.asctec.de/

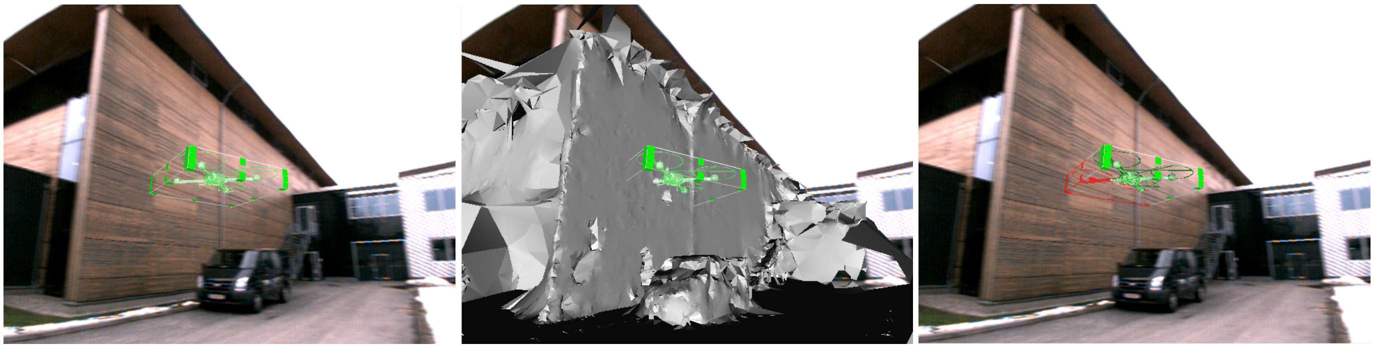

Fig. 2. Physical depth cues for visualizing the planned position of a MAV. While a simple overlay of the impostor shape representing the MAV (Left) does not give enough information about the spatial relationship between vehicle and building, an approximate mesh representation (Middle) used for depth testing highlights occluded object parts (Right).

pitch due to auto balancing. This means a single waypoint consists of longitude, latitude, height and yaw. The waypoint describes where the aerial vehicle will move next. Other information contained in a waypoint are the holding time (how long the vehicle waits at a position) and maximum speed with which it is moving. For convenience, a pathlist manages a list of all planned waypoints for one flight session. The vehicle will process the items of the list in order as they are stored in the list.

While the aerial vehicle itself cannot manipulate pitch and roll, the camera is mounted on a steerable frame providing more flexibility for the camera viewing angles. The camera control parameters give access to the pitch and roll of the steerable frame.

Traditionally, both types of information are processed by either automatic flight path generation tools or map-based MAV control interfaces. In the following, we will describe how the information can be processed and visualized by an AR system.

3.1 Visualization of Waypoints and Pathlists

Before the data provided by a MAV can be visualized in an AR overlay, it has to be converted from geo-referenced data into a graphical representation. For this purpose, we create different 3D geometries that represent the flight-relevant information. While a 3D MAV geometry provides the actual shape of the vehicle (Figure 2), abstract representation, such as spheres representing positions of waypoints, can illustrate inaccuracies of the geo-referenced locations using the radius of the sphere. In the FlyAR visualization, a single sphere represents either one planned waypoint or the current location of the vehicle, similar to the representation of geo-referenced sensor data by White et al. [26]. The center of the sphere is described by longitude, latitude and height. Color mapping is used to represent different data types with different colors. For instance, dark blue represents the current state of the MAV (Figure 3), light blue shows flight paths created by an external application and purple indicates waypoints created on-site with the FlyAR interface. A pathlist is represented as a set of spheres that are connected by a line indicating their order.

Furthermore, it is important for the user to understand the orientation of the vehicle and the camera. In particular, if the vehicle is far away, the direction where the camera is looking is often not visible to the human eye. We visualize this direction by using a rectangular stick starting in the middle of the waypoint and pointing in viewing direction of the camera (Figure 3, Right). These visualization techniques for flight management data give the user first insights about the spatial relationship of planned waypoints, current positions and viewing directions within the physical environment.

Unfortunately, such a simple overlay of flight-relevant data lacks pictorial depth cues that are important to understand the spatial arrangement in detail. By nature, the depth perception of flying objects that are not within the near field is complicated. As described by Cutting, some depth cues, such as *convergence* and *accommodation*, are only available in the *personal space* (up to 1.5 m from the user's head, which we will refer to as near field in the following) [3]. Others, such as *motion perspective* are only working in the *action space* defined by Cutting as the area up to 30 m from the user (which we will refer to as mid field). Objects that are further away than 30m, as it is often the case for MAV navigation, are located in the *vista space* as defined by Cutting (we will refer to as far field). In the vista space cues such as *occlusion*, *relative size* and *height in visual field* (objects that are further away are closer to the horizon) are assumed to provide depth information. Other sources such as *aerial perspective* will be neglected in our discussion, since it is assumed to work for distance over 100 m, which too far for MAV flight supervision. While occlusion and relative size provide only ordinal or relative measurements, in their work from 1995 Cutting and Vishton assumed the cue *height in visual field* to have the potential of yielding absolute distances, if a set of assumptions is fulfilled [4]. These assumptions include that the ground plane in the scene is nearly planar and objects have their base on this ground plane. This can explain why it seems more complicated to judge the depth of flying objects at a distance, since in this case these assumptions cannot be fulfilled.

Table 1. Overview of visualization methods.

	Method	Purpose
Physical Cues	Occlusion Culling	Occlusion Cues
	Alpha Blending	Occlusion Cues
	Highlighting	Warning
Virtual Cues	Virtual Shadow	Depth, world relationship
	World-Centric Connection	Depth, world relationship
	Object-Centric Connection	Between waypoints
	User-Centric Connection	Relationship to user

These observations emphasize the two main perceptual issues for visualizing flight-relevant data in AR. Firstly, the visualization should preserve as many natural depth cues as possible. This is often a challenge in simple AR overlays, in particular for the preservation of occlusion cues [1]. Secondly, natural pictorial cues are often not enough to understand the absolute depth of flying objects in the far field, since some pictorial cues are not fully working (such as height in visual field). Thus, the AR visualization should support depth perception by including additional graphical hints that are naturally not available.

In the following, we will discuss how we 1) maintain **physical depth cues** such as occlusion to support the understanding of spatial relationships as well as critical locations of the aerial vehicle in the physical world and 2) integrate additional graphical hints, so-called **virtual cues**, to support the spatial perception of flying virtual objects representing the aerial vehicle. In Table 1, we give a classification of the proposed visualization techniques and their purpose.

3.2 Physical Depth Cues

Several physical depth cues are already given by the rendering, such as the relative size. However, for providing the strongest cue, occlusion, the composition techniques in AR have to be adapted [1]. For

Fig. 3. Graphical hints for supporting the spatial understanding of the MAV (Asctec Falcon 8 in black). (Left) Sphere representation of the waypoint and its shadow projection on the ground. (Middle) Connection line combined with shadow. (Right) User-centric graphical cue indicating the distance of the waypoint to the user's location.

a convincing occlusion management, the composition method needs to know the depth order of virtual and physical objects in the scene. This means in our case, we need at least an approximate virtual representation of the physical environment for depth testing. Based on this depth information, we integrated three different methods for providing occlusion cues: 1) a complete occlusion culling, 2) alpha blending and 3) highlighting of occluded regions. While complete occlusion culling excludes occluded fragments from the rendering and creates a realistic occlusion effect, alpha blending allows for an X-Ray view inside the occluding object. Additionally, by highlighting these critical positions, the AR visualization allows the user to quickly identify critical positions.

Using this approach, we highlight occluded waypoint information (Figure 2, Right) based on a mesh representation (Figure 2, Middle) of the working area. This allows us to provide depth cues, but also to indicate if a waypoint of interest is moving out of the user's view.

3.3 Virtual Cues

Unfortunately, natural depth cues often do not provide enough absolute depth information for flying objects in the far field. In particular, some natural cues do not fully work, because flying objects do not fulfill certain requirements. For instance, height in visual field only works for objects that are connected to a ground plane [3]. In order to address the problem of too little depth information and to restore natural cues, we introduce a set of additional graphical hints. Thereby, we integrated cues for supporting the comprehension of the spatial relationship 1) between waypoints and the physical world, 2) between subsequent waypoints, 3) between waypoints and the user's point of view, 4) as well as interactive measurements in the scene.

Spatial relationship between waypoints and physical world As mentioned above, one of the main issues for understanding the spatial arrangement of a flying object is the missing connection between the physical ground and the object. This reduces the effectivity of the cue *height in visual field*. In order to restore the cue's ability to provide absolute measurements, we create *virtual connections* between the flying objects and the ground. These virtual connections are represented by rectangular objects starting at the location of the waypoint and ending on the ground (Figure 3, Middle). A requirement to create the virtual connections is either information about the ground plane or the height above mean sea level at the location of the waypoint. We further support depth estimation by adding height indicators such as a metric scale to the connection lines. For this purpose, we map a texture with a metric scale onto the virtual connection. One of the main disadvantages of such a metric scale texture is reduced readability at higher distances. To overcome this problem, we experimented with different scale sizes and integrated dynamic textures that depend on the distance to the user.

Another depth cue that has been proposed to be helpful for the depth perception of flying objects in AR are shadows. While Wither et al. included an artificial shadow plane to visualize shadows of virtual flying objects [28], we use the ground plane of the physical environment as a shadow plane. For this purpose, we create a flat rectangular object that is located at the position of the waypoint, but at the height of the ground plane. This graphical hint appears as a virtual shadow of the waypoint (Figure 3, Left). We decided for a rectangular shape as shadow representation to reflect the shape of the virtual connection.

Spatial relationship between subsequent waypoints For path planning as well as for supervising a flight session, not only the position of single waypoints is important, but also the spatial relationship between waypoints. By adding *virtual connections* with metric scales between waypoints, we support the user in estimating the distances between single waypoints. In this case, each virtual connection starts at the location of one waypoint and ends at the location of its successor (Figure 5, Right).

Spatial relationship between waypoints and user For the supervision of a flight session it is also important to have information about the distance between the vehicle and the position of the supervising user. Therefore, we offer a third group of additional graphical hints that provide user-centric depth cues. The user-centric cues consist of a virtual connection from the user's position on the ground, the projection of the waypoint on the ground plane and the waypoint location itself (Figure 3, Right).

Interactive Measurement Hints During the planning of new waypoints the distance between physical obstacles and the newly planned waypoint is interesting to the user as well. For this purpose, we integrated an interactive method for selecting obstacles in the physical scene and visualizing the distance by using a virtual connection between waypoint and obstacle. To realize this, we assume that an approximate 3D representation of the obstacle is available.

4 SYSTEM

We integrated the aforementioned visualization techniques for flight management into an interactive mobile AR system. For this purpose, we need 1) to access data that is relevant for flight management, 2) a mobile AR client that is capable of exchanging and displaying this data and is 3) registered in relationship to the data. To achieve high flexibility and reliable communication, we use ROS (Robot Operating System) for exchanging flight relevant data between the AR client, the MAV and other flight management applications (Figure 4).

4.1 Registration

For the registration of the AR setup, we implemented two different methods. While the first one uses a set of additional sensors, the second one uses only the camera image for registration. The sensor-based registration combines high accuracy GPS, IMU and vision-based tracking and achieves accuracies in the subangle and centimeter

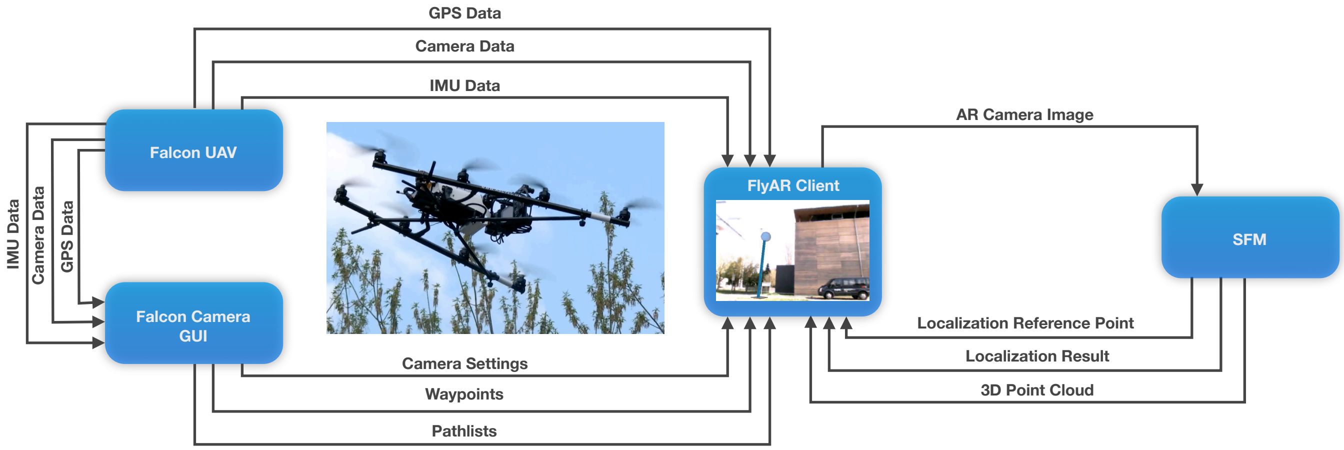

Fig. 4. FlyAR system overview. (Left) Falcon MAV and Falcon Camera GUI publish flight-relevant information such as live GPS data or planned flight paths. (Middle) The AR client registers to the corresponding topics and receives the information. Furthermore, the AR client publishes camera images and uses their localization matrices for visualizing the flight-relevant data in an AR overlay. (Right) The localization and reconstruction client (SFM) receives and localizes the camera images from the AR client and publishes the localization result.

range [22] under ideal conditions (no shadowing, no magnetic distortions).

However, if an accurate geo-referenced 3D model of the environment, such as a 3D point cloud, we use the 3D point cloud of the environment and the video image data as input for localization and tracking. This method starts with the localization step that is done remotely over the network. It uses the geo-referenced 3D point cloud to compute an initial global pose from the current camera image, similar to the methods of Ventura et al. [25, 24]. The localization step is not in real-time, but takes less than a second. For real-time tracking, the AR client initializes a local tracking model using the initial pose, the 3D point cloud data and the 2D image data of the initial camera frame. This local tracking model is then used to match new camera frames to the 3D point cloud information. In contrast to the methods of Ventura et al., our tracking method addresses the problem of time- or season-dependent illumination changes by combines the remote 3D point cloud data with the appearance information of the initial camera frame.

The assumption of having a 3D point cloud of the area of interest available for tracking for aerial 3D reconstruction sounds like a chicken-egg problem on the first sight. Nevertheless, this assumption is valid in many cases for the workflow of creating aerial 3D reconstructions in construction environments. For instance, often after capturing a first 3D reconstruction, the construction site staff identifies areas of interest that should be captured with a higher level of detail and requires the MAV to navigate closer to these building parts. Another example having a 3D point cloud available for tracking is construction site documentation where a 3D model of the former status exists and should be updated [29]. Even a geo-referenced 3D reconstruction based on terrestrial views can be used as data input. Therefore, it is important that the locations where a registration should be performed, are captured in the 3D model. Additionally, by integrating newly localized images into the 3D reconstruction, the localization area is continuously extended.

4.2 Physical World Context Data Sources

For registration and for visualization, we need information about the physical world, such as the 3D point cloud or mesh information.

Point Cloud We create a sparse 3D point cloud of the environment by capturing a set of aerial images of the area of interest and using SfM [14]. The sparse geometry can already be used for initializing the tracking model, but contains only limited data for visualization. State-of-the-art methods allow for creating a semi-dense point cloud [8] (Figure 1, Left). GPS information from the MAV helps to reduce computation time and to create geo-referenced data.

Mesh For inspecting occluded waypoints, we need dense depth information about the environment. Geo-spatial databases such as geographic information systems (GIS) already contain dense depth information about our physical environment. For areas of interest where this data is not available or too sparse, we use the 3D point cloud data as input for a mesh creation. Meshes of a 3D point cloud data can be computed by combining a Delaunay triangulation with global optimization methods [18] and can even be computed in real-time allowing for online-feedback [11].

Interactive Input of Obstacles For dynamic environments, such as construction sites, a tool for on-site input of obstacles is often required. For this purpose, we provide the user tools to interactively insert impostors of physical obstacles to the scene to inspect possible conflicts. By clicking on the ground plane in the AR view, the user selects a 3D point in space where an impostor geometry is placed. For selecting the point in 3D we need information about the ground plane given by either a digital terrain model (DTM) or given by the mesh data described in the previous paragraph.

4.3 Mobile AR Client

For testing the approach outdoors, we use a ruggedized tablet PC (Motion J3400) in combination with a camera (VRmFC-6 PRO). The camera is used for registration and video capturing. Additionally, the setup is equipped with an IMU and a GPS receiver to support test scenarios where no 3D point cloud is available for model-based tracking.

5 APPLICATIONS

The proposed FlyAR interface brings different aspects of the flight management process on-site. In this section, we describe them more in detail.

5.1 On-site inspection of planned flight paths

If a previously computed pathlist is published over the network, the AR client is notified and all waypoints are converted into graphical objects (sphere or 3D model). The waypoints are published as geo-referenced data in WGS84 format. Due to the single precision ability of OpenGL, we have to convert the geo-referenced information into a local coordinate system before we can display them. For this purpose, we define a reference point in WGS84 that is used as the center point of the local coordinate system. After the conversion, all information is available as single precision coordinates and can be used for rendering and is combined with the visualization techniques presented in Section

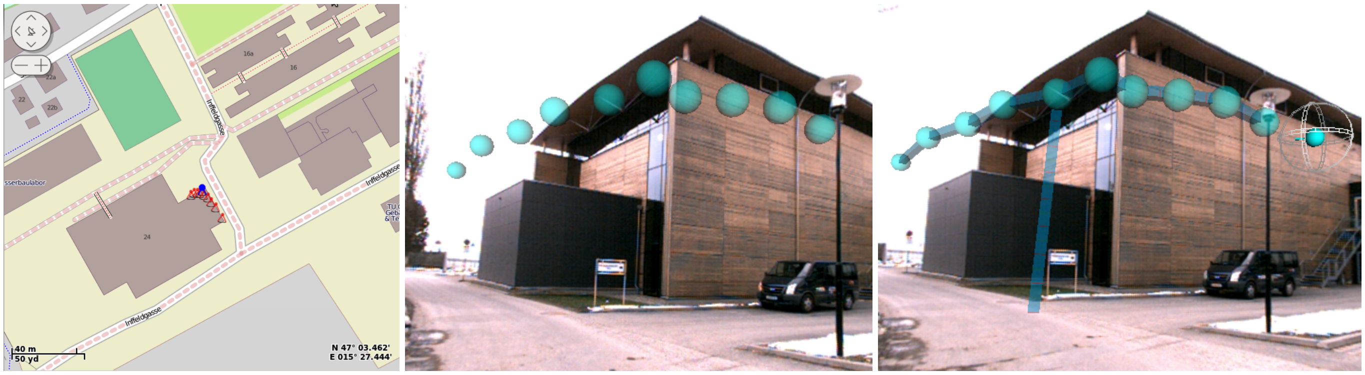

Fig. 5. Flight path visualization. (Left) Planned flight path in 2D map interface. Planned waypoints are marked with red. The current location and orientation of the aerial vehicle is marked with blue. (Middle) Planned flight path simply overlaid on imagery of the physical environment. (Right) Planned flight path visualization enriched with additional graphical hints showing the height and the distances between single waypoints. An interactive spherical modification tool allows manipulating the orientation of a waypoint.

3.2, for instance showing virtual connections between all waypoints (Figure 5, Right).

5.2 On-site AR flight planning

With the on-site flight planning mode, the user can interactively create new waypoints or modify existing ones. A manipulation tool allows either changing the position of the waypoint or the orientation of the steerable camera frame. For manipulating the position of the waypoint, we use a box geometry that translates the waypoint if the users clicks and drags its sides. For changing the orientation, the user has to click on a spherical representation around the waypoint and drag along the axis selected for manipulation (Figure 5, Right). If the user wants to add a new waypoint to the planned path, a button in the GUI confirms the waypoint. At this point, the geometric information is converted from the local coordinate system to the geo-referenced format and stored in an XML file. Finally, the new waypoint is highlighted as already stored. The XML file can then later be loaded into to the 2D map GUI and sent to the MAV.

5.3 Live flight supervision

In the live flight supervision mode, the AR client displays all flight specific information published by the aerial vehicle, such as GPS data, IMU data and camera settings (visualized in Figure 4. The AR client converts this data again into the local coordinate system and visualizes it with graphical 3D representations and adds visual cues. For the live flight supervision, we visualize the current state of the MAV as well as previous positions and positions where camera images were captured. Additionally, we visualize planned waypoints to understand the next steps and to avoid collisions with obstacles.

6 USER EVALUATION

To gain first feedback we conducted a user evaluation. The general goal was to verify the usefulness and create understanding if people benefit from the proposed system and the visualization techniques.

6.1 Study design

Task and Procedure The task of our user study focused on estimating the position of a MAV flying at a static position within a known environment. We therefore asked each participant to observe the flying MAV either without support from the FlyAR system or while using the FlyAR system and to mark the MAV's position in a map for six different positions visible from a single location. We did not considered a moving MAV in our study, as it would add many additional factors to our study. By using the proposed study design, we can focus on the harder part of the overall problem: Estimating the drone without the effect of motion parallax. Latter is likely to add additional depth cues easing the problem for the user. We further argue that the specific study design is still highly applicable to the general problem of semi-automatic flight control. Not only was it proposed by several of our professional drone pilots as one of the main problems when controlling a drone but also the industrial scenario of this project often requires a very precise navigation to defined positions (e.g., for capturing details of the construction site). This happens often at very low speed with and only very small movements in the final approach. Consequently, in these scenarios the effect of motion parallax is very small, especially when seen from larger distances. In summary, we argue that the conducted study only explores certain aspects of the full problem of estimating the position of a drone but can still be seen as a first important step towards semi-automatic flight control of MAV.

The procedure was as follows: After a short introduction to the study, the participants were introduced to the test environment. We made sure that the participants were already familiar with the test environment. The participants had the chance to see the visualization during a short demo flight of the MAV. During the demo, the participants had time to adjust the tablet PC and ask remaining questions. We also made sure that participants had a mental model of mapping their environment to the provided map and vice versa.

The actual study started once no remaining questions were left, by navigating the MAV to the first of the six positions. We asked the participants to turn around so that they could not see the approaching MAV, to avoid the usage of depth cues caused by motion parallax. Once the MAV stabilized at the final position, we asked the participant to turn around and to estimate the position of the MAV using the provided questionnaire. In this questionnaire the participant needed to mark the position on the map. The participants were further asked in the questionnaire to fill out how easy it was to estimate the position and their confidence with the position estimate on a 7-point Likert scale. We were limited in flight time to approx. 10 min given by the battery of the MAV. We therefore were limited in time for discussions and observations for collecting qualitative feedback of the participants. To overcome this problem, we integrated a thinking-aloud protocol into the study. We encouraged the users to speak out their thoughts while conducting the position estimate and filling out the questionnaire which we recorded using a small body-worn Mecam video camera[2]. One investigator made further notes while conducting the study. After filling out the questionnaire for the particular position, we asked the participants again to turn around so that we could navigate the MAV to the next position.

We continued with this approach for the remaining positions (six in total), however, for three out of the six positions the participants were allowed to use our AR visualization displayed on the tablet PC as supporting the position estimation while for the remaining three

---

[2] http://www.mecam.me

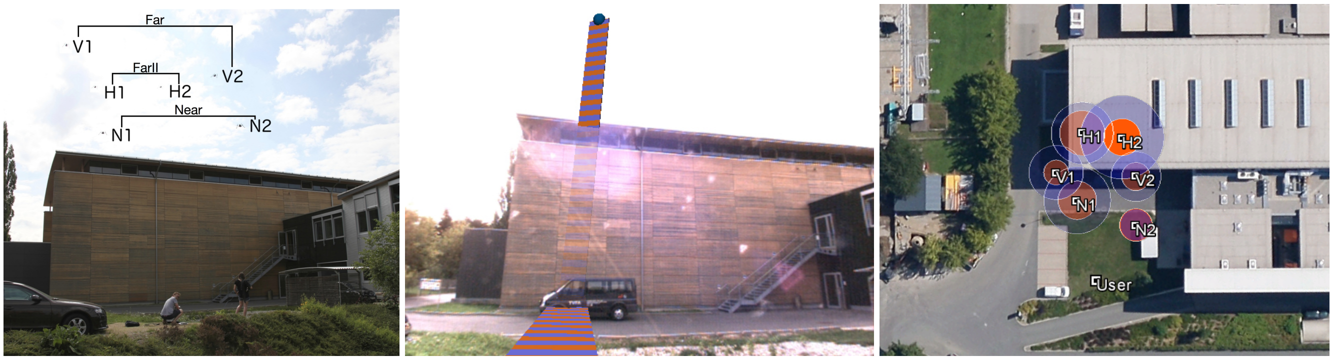

Fig. 6. User study. (Left) Six positions of the MAV during the study. There are two positions in mid distance to the participants (U-N1: 20m, U-N2:18m), two positions in far distance (U-V1: 28m U-V2: 29m) and two positions even at further distance (U-H1:33m, U-H2:36m). This picture is a photo-montage by blending several images taken from the same position into one. The user study was conducted from the location visible in the foreground. (Middle) Example visualization for position H1 with the FlyAR system. (Right) Results from the localization estimation by participants. Blue transparent circles indicate the average error for simply observing the MAV. Orange circles indicate the average error using the FlyAR system.

position they had to mark the position on the map without AR support by only looking to the MAV. The order of positions and supporting techniques used (with AR or without support) was obtained using a randomized Latin square in order to compensate for learning effects.

We concluded the study with a short final questionnaire about usability and applicability for both conditions using 7-point Likert scales and collected qualitative feedback as part of a semi-structured interview. The interview had fixed questions regarding the environmental conditions, the hardware setup, and the particular visualization and was followed by other questions resulting from observations.

Experimental Platform Within our study we relied on a professional MAV (Asctec Falcon 8) (see Figure 1, Middle). This MAV is an octocoper with roughly the dimensions of 65cmx55cm and a maximum weight of 2,2 kg. The MAV was controlled by a professional pilot assisted by a flight planning program and automatic position control, allowing us to precisely keep the MAV at the defined position. The AR visualization was displayed on the mobile AR setup that was mounted on a tripod with a ball head allowing free adjustment and steering of the tablet when in use. Before we started with the user evaluation, we made adjustments to the visualization based on feedback from a professional MAV pilot. We chose a fixed color pattern for the scale texture that is unlikely to interfere with natural colors from the physical surroundings and we choose a standard width of the virtual connections of 2 meters.

Hypotheses Within the study we were interested in the benefit of using the FlyAR system over solely observing the MAV. Thus, we had the following three hypotheses:

- H1: The localization estimation will be less error prone when using the FlyAR system compared to solely observing the MAV.
- H2: The localization task will be perceived to be easier while using the FlyAR system and participants will be more confident about their estimates.
- H3: The perceived usefulness of the FlyAR system is higher than solely observing the MAV.

Participants We invited 14 participants to take part in this experiment (2 female, 12 male, age ranging from 22 to 38, with different backgrounds, ranging from students over lab members to external participants). For one participant we had to cancel the experiment due to technical issues concerning the data exchange with the MAV, for another participant we had to exclude the results since a wrong flight sequence was used for the MAV. The experience with AR of the participants ranged from not familiar at all to very familiar. A lot of the participants had no or nearly no experiences with MAVs (8/14). We used a repeated measure design for the study. Each participant performed a localization estimation for 6 locations on the test site under two conditions (C_1 : solely observing MAV, C_2 : with AR). These 6 locations were subdivided into 3 distance categories (mid (approx. 20m), far (approx. 30m), farII (approx. 35m)).

6.2 Results

Quantitative Results For each participant, we computed the error between the ground truth location and the location indicated by the participant on the map. Based on these measurements, we computed the average error per participant for both conditions. The results showed that the participants performed better under condition C_2 using the FlyAR system than solely observing the MAV ($M_{MAV} = 5.11$ vs. $M_{AR} = 3.01$, Boxplot in Figure 7, Left). A Wilcoxon signed-rank test showed that there is a significant effect of the condition to the average error (Z = 3.06, p-value = 0.0022, r = 0.625).

In order to understand if the effect exists for all tested positions (N1, N2, V1, V2, H1, H2; as visualized in Figure 6, Right), we also computed the estimation error based on using the AR system and solely observing the MAV. The estimation error using AR was smaller for all positions (C_2 (N1: M_{MAV}=5.89, M_{AR}= 3.45; N2: M_{MAV}= 2.91 M_{AR}= 2.81; V1: M_{MAV}= 4.76 M_{AR}=2.20; V2: M_{MAV}=4.42 M_{AR}= 2.54; H1: M_{MAV}= 5.55 M_{AR}= 4.00; N2: M_{MAV}= 7.43 M_{AR}= 3.32, Figure 6), Right). However, the difference between both conditions for position N2 was very small. A Mann-Whitney U test indicated significant differences between both conditions for all locations except N1 (compare Table 2).

Table 2. Localization estimation error for single positions.

Position	Result	Z score	p value	Pearson's r
N1	$MAV > AR$	-1.9215	0.055	0.55
N2	$MAV > AR$	0.0	1.0	0.0
V1	$MAV > AR$	-1.92	0.055	0.55
V2	$MAV > AR$	-1.92	0.055	0.55
H1	$MAV > AR$	-2.24	0.024	0.65
H2	$MAV > AR$	-2.08	0.037	0.60

Subsequent to each position's estimation task, the participants were asked to rate how easy this task was and how confident they felt with their position estimate. The results from this questionnaire showed that the participants experienced the location estimation as being easier while using the FlyAR system, compared to solely observing the flying MAV (M_{MAV}=4.5 M_{AR}=5.42). Furthermore, they felt rather confident with their estimate (M_{AR}=5.30) using the FlyAR system, compared to being rather undecided about the confidence while observing the MAV (M_{MAV}=4.36). These results are also visualized in Figure 7. A Wilcoxon signed-rank test showed that there is a significant effect of the condition to the perceived confidence (Z = 2.47, p-value = 0.0134, r = 0.505). The test also showed a significant effect of the condition to the perceived easiness (Z = 2.17, p-value = 0.03021 , r = 0.442).

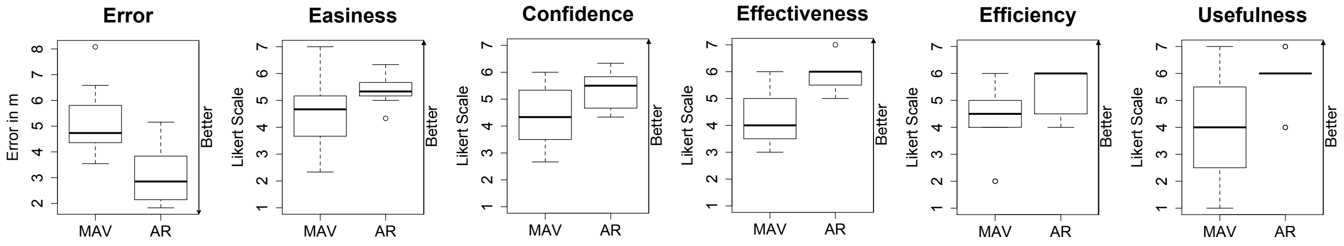

Fig. 7. Results. From Left to Right: Boxplots for average localization error, perceived easiness, confidence of participants in localization estimate, general effectiveness, general efficiency and general usefulness.

In the final questionnaire, we asked the participants about the perceived effectiveness, efficiency and usefulness. For all these questions, the AR interface outperformed the MAV condition with M_{MAV} = 4.33 M_{AR} = 5.83 for effectiveness, M_{MAV} = 4.25 M_{AR} = 5.33 for efficiency, and M_{MAV} = 3.90 and M_{AR} = 5.82 for usefulness respectively (Figure 7). It seems that the participants were rather undecided for all these questions while solely observing the MAV. For these questions we performed a Wilcoxon signed-rank test as well showing significant effects of the condition to the questions (Effectiveness Z = 2.98, p-value = 0.0029, r = 0.609, Efficiency: Z = 2.22, p-value = 0.0265 r=0.452, Usefulness: Z = 2.02, p-value = 0.0432, r =0.431).

Qualitative Results The observation from the recorded video material together with the notes made by the investigator during the trials and in the semi-structured interviews gave further insights. Overall the participants of our study all acknowledged the general problem of estimating the position of the MAV, stating that it was in general quite challenging to identify the position, in particular if the AR setup could not be used. Regarding the environmental conditions all but two participants stated that the light was not affecting the study (e.g., bright sky, reflection on the screen). Two participants had small problems, one told us after the study that they were very sensitive to sun-light and even though sun glasses were provided if needed, there were remaining problems accommodating for the brightness difference between the screen and the sky. The other participant stated that for one position one had to nearly look into the sun so the participant exclusively relied on the AR visualization and felt a bit less confident compared to cases when both the AR view and normal view could be used. Even though all except three participants noticed that the MAV was moving slightly due to the wind, they all said it was not an issue for the study as it was stable enough for the task. The majority replied that the maximum movement is less than 50cm with only one participant stating that at one position it was not more than 1m, while less at the other positions. All participants stated that the MAV immediately corrected the position.

All participants had no problems with the general setup and the adjustment of tablets position using the ball head. However, one participant stated their wish for a wider angle camera so no adjustment of the tablet is needed as the MAV and the visualization will always be in view.

Some participants stated that they would have liked other colors used in the AR visualization but that this has not affected the study and is only for aesthetic reasons. However, when asked which color or color combination they would prefer, they noticed the visibility problem (e.g., being clearly visible in-front of the sky or buildings) and could not gave a better color scheme. All participants stated that the size of the visualizations (e.g., width of the lines) was perfect. However, one participant stated that he usually only focused on the position where the augmented line crossed the ground. In that case he sometimes was not aware if the augmentation still aligned with the physical world (the MAV). He stated that for these cases it could be helpful if other landmarks could be augmented to verify correct alignment with the physical world.

6.3 Discussion

The results of our study confirmed our hypotheses and indicate that our approach of AR supported MAV navigation improves the position estimate and consequently the navigation of MAVs. All participants achieved better position estimates when using our AR visualization compared to solely relying on observations.

While the results of the study suggest a possible effect of the distance to the MAV by showing that especially in cases where the MAV is further away, the users do especially benefit from the AR cues, we showed that even at close positions the position estimate is more accurate. We argue that for the position N2, where this result was visible but not significant, we probably need a larger number of participants to sufficiently investigate the effect and significance. The difference in the results for these positions achieved using pure observation or using the AR visualization was given but too small to guarantee statistical evidence given the number of participants in our study.

The result was also reflected in the given feedback for each position showing that the participants not only said it is easier to estimate the position using our system but also felt more confident when using our system as well as in the final questionnaire stating that they could more effectively and efficiently solve the given task of estimating the MAVs position using the AR visualization.

The focus of our study was the evaluation of our system and the used visualization, however, related tasks such as navigation often showed a gender effect, which we were not particularly interested in. We further would also not be able to statistically analyze it given the current, non-gender balanced groups. However, we do not want to exclude the possibility.

The issues reported by the users were mainly about details of the particular visualization in term of general aesthetics such as the used colors on contrast, which could be changed in future version to automatically adapt to the current context (user or environment). The feedback of the study showed also that while the participants did not have problems with the stability of the MAVs position but also indicated that live feedback about pose stability could be further increasing their confidence.

7 CONCLUSION AND FUTURE WORK

In this paper, we showed how AR can support the flight management process for aerial vehicles. We proposed FlyAR, an AR interface that superimposes the user's view with flight specific information for flight path planning as well as for the supervision of flight sessions. By overlaying planned waypoint data, the user can interactively inspect a planned path for a flight session and compare it to the on-site situation. The interface provides functionality to plan flight sessions directly on-site and supervise flight sessions. Furthermore, we introduced a set of visualization techniques that support depth perception by providing additional depth cues for flying objects. For this purpose, we integrated physical as well as virtual graphical hints into the visualization.

Furthermore, we conducted a user study and demonstrated the positive effect of using the FlyAR system for live flight supervision on the error in position estimates, on the user's confidence and the perceived easiness of estimating the position of a MAV.

Experiences that we made during the user evaluation and working with the system, is that one of the biggest issues is if the users rely on the accuracy of the system and the sensor data from the MAV is inaccurate, for instance due to shadowing effects on the GPS receiver. To address this problem, we plan to integrate visual tracking, as we use it already for the localization of the AR device, into the localization of the MAV.

Further feedback from a user with experience in supervising autonomous flight sessions inspired us to extend our visualization techniques with further information, such as highlighting the area covered by the camera shot in the physical world or integrating live feedback for SfM image acquisition [12].

ACKNOWLEDGMENTS

The authors wish to thank Raphael Grasset and Dieter Schmalstieg for valuable feedback. This work was funded by the Austrian Research Promotion Agency (FFG) FIT-IT project Construct (830035) and by the EC 7th Framework Programme (grant 611526).

REFERENCES

[1] D. E. Breen, R. T. Whitaker, E. Rose, and M. Tuceryan. Interactive Occlusion and Automatic Object Placement for Augmented Reality. *Computer Graphics Forum*, 15(3):11–22, Aug. 1996.

[2] F. D. Crescenzio, G. Miranda, F. Persiani, and T. Bombardi. A First Implementation of an Advanced 3D Interface to Control and Supervise UAV (Uninhabited Aerial Vehicles) Missions. *Presence: Teleoperators and Virtual Environments*, 18(3):171–184, June 2009.

[3] J. E. Cutting. How the eye measures reality and virtual reality. *Behavior Research Methods, Instruments, & Computers*, 29(1):27–36, 1997.

[4] J. E. Cutting and P. M. Vishton. Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth. *Handbook of perception and cognition*, 5:1–37, 1995.

[5] A. Dey, A. Cunningham, and C. Sandor. Evaluating depth perception of photorealistic mixed reality visualizations for occluded objects in outdoor environments. In *ACM Symposium on Virtual Reality*, 2010.

[6] J. Engel, J. Sturm, and D. Cremers. Camera-based navigation of a low-cost quadrocopter. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2815–2821. IEEE, Oct. 2012.

[7] F. Fraundorfer, L. Heng, D. Honegger, G. H. Lee, L. Meier, P. Tanskanen, and M. Pollefeys. Vision-based autonomous mapping and exploration using a quadrotor MAV. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4557–4564. IEEE, Oct. 2012.

[8] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Towards Internet-scale multi-view stereo. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1434–1441, June 2010.

[9] A. Goektogan and S. Sukkarieh. An Augmented Reality System for Multi-UAV Missions. In *SimTecT*, 2005.

[10] S. Hashimoto and A. Ishida. TouchMe: An Augmented Reality Based Remote Robot Manipulation. In *Proceedings of ICAT2011*, 2011.

[11] C. Hoppe, M. Klopschitz, M. Donoser, and H. Bischof. Incremental Surface Extraction from Sparse Structure-from-Motion Point Clouds. In *British Machine Vision Conference*, pages 1–11, 2013.

[12] C. Hoppe, M. Klopschitz, M. Rumpler, A. Wendel, S. Kluckner, H. Bischof, and G. Reitmayr. Online Feedback for Structure-from-Motion Image Acquisition. In *Procedings of the British Machine Vision Conference 2012*, pages 70.1–70.12, 2012.

[13] C. Hoppe, A. Wendel, S. Zollmann, K. Pirker, A. Irschara, H. Bischof, and S. Kluckner. Photogrammetric Camera Network Design for Micro Aerial Vehicles. In *Computer Vision Winter Workshop*, 2012.

[14] A. Irschara, V. Kaufmann, and M. Klopschitz. Towards fully automatic photogrammetric reconstruction using digital images taken from uavs. In *ISPRS - Advancing Remote Sensing Science*, 2010.

[15] K. Ishii, S. Zhao, M. Inami, T. Igarashi, and M. Imai. Designing laser gesture interface for robot control. In *INTERACT '09 Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction: Part II*, pages 479–492, 2009.

[16] S. Julier, D. Brown, M. Livingston, and J. Thomas. Adaptive synthetic vision. In *Proceedings of SPIE - The International Society for Optical Engineering*, Oct. 2006.

[17] S. Kasahara, R. Niiyama, V. Heun, and H. Ishii. exTouch: spatially-aware embodied manipulation of actuated objects mediated by augmented reality. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction*, 2013.

[18] P. Labatut, J. Pons, and R. Keriven. Robust and efficient surface reconstruction from range data. *Computer Graphics Forum*, 28(8), 2009.

[19] M. Livingston, Z. Ai, K. Karsch, and G. O. Gibson. User interface design for military AR applications. *Virtual Reality - Special Issue on Augmented Reality*, 15(2-3):175–184, Dec. 2011.

[20] A. Nawab, K. Chintamani, D. Ellis, G. Auner, and A. Pandya. Joystick mapped Augmented Reality Cues for End-Effector controlled Teleoperated Robots. *Virtual Reality Conference*, pages 263–266, 2007.

[21] M. Quigley. Semi-autonomous human-UAV interfaces for fixed-wing mini-UAVs. In *Proceedings of IROS*, 2004.

[22] G. Schall, D. Schmalstieg, and S. Junghanns. VIDENTE-3D Visualization of Underground Infrastructure using Handheld Augmented Reality. *Geohydroinformatics-Integrating GIS and Water Engineering" CRC Press/Taylor and Francis Publisher: CRC*, 1:1–17, 2010.

[23] J. Stöcklein, P. Pogscheba, C. Geiger, and V. Paelke. MiReAS. In *Proceedings of the 7th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa - AFRIGRAPH '10*, page 27, New York, New York, USA, June 2010. ACM Press.

[24] J. Ventura, C. Arth, G. Reitmayr, and D. Schmalstieg. Global Localization from Monocular SLAM on a Mobile Phone. In *Submitted to VR 2014*.

[25] J. Ventura and T. Hollerer. Wide-area scene mapping for mobile visual tracking. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2012)*, number November, pages 3–12, 2012.

[26] S. White. Interaction with the Environment: Sensor Data Visualization in Outdoor Augmented Reality. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2009)*, pages 5–6, 2009.

[27] J. Wither, S. Diverdi, and T. Hollerer. Using aerial photographs for improved mobile AR annotation. In *International Symposium on Mixed and Augmented Reality*, pages 159–162. IEEE, Oct. 2006.

[28] J. Wither and T. Hollerer. Pictorial depth cues for outdoor augmented reality. In *Proceedings of the Ninth IEEE International Symposium on Wearable Computers (ISWC '05)*, 2005.

[29] S. Zollmann, D. Kalkofen, C. Hoppe, S. Kluckner, H. Bischof, and G. Reitmayr. Interactive 4D overview and detail visualization in augmented reality. In *International Symposium on Mixed and Augmented Reality*, 2012.

[30] S. Zollmann, G. Schall, S. Junghanns, and G. Reitmayr. Comprehensible and Interactive Visualizations of GIS Data in Augmented Reality. *Advances in Visual Computing*, pages 675–685, 2012.