Expert Systems With Applications 82 (2017) 100–114

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Markerless tracking system for augmented reality in the automotive industry

João Paulo Lima[a,b,*], Rafael Roberto[a], Francisco Simões[a], Mozart Almeida[a], Lucas Figueiredo[a], João Marcelo Teixeira[a,b], Veronica Teichrieb[a]

[a] Voxar Labs, Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), Av. Jornalista Anibal Fernandes, s/n - Cidade Universitária, 50740-560 - Recife, PE, Brazil

[b] Departamento de Estatística e Informática (DEINFO), Universidade Federal Rural de Pernambuco (UFRPE), Rua Dom Manoel de Medeiros, s/n - Dois Irmãos, 52171-900 - Recife, PE, Brazil

ARTICLE INFO

Article history:
Received 30 April 2016
Revised 6 February 2017
Accepted 25 March 2017
Available online 29 March 2017

Keywords:
Tracking
Augmented reality
Automotive sector

ABSTRACT

This paper presents a complete natural feature based tracking system that supports the creation of augmented reality applications focused on the automotive sector. The proposed pipeline encompasses scene modeling, system calibration and tracking steps. An augmented reality application was built on top of the system for indicating the location of 3D coordinates in a given environment which can be applied to many different applications in cars, such as a maintenance assistant, an intelligent manual, and many others. An analysis of the system was performed during the Volkswagen/ISMAR Tracking Challenge 2014, which aimed to evaluate state-of-the-art tracking approaches on the basis of requirements encountered in automotive industrial settings. A similar competition environment was also created by the authors in order to allow further studies. Evaluation results showed that the system allowed users to correctly identify points in tasks that involved tracking a rotating vehicle, tracking data on a complete vehicle and tracking with high accuracy. This evaluation allowed also to understand the applicability limits of texture based approaches in the textureless automotive environment, a problem not addressed frequently in the literature. To the best of the authors' knowledge, this is the first work addressing the analysis of a complete tracking system for augmented reality focused on the automotive sector which could be tested and validated in a major benchmark like the Volkswagen/ISMAR Tracking Challenge, providing useful insights on the development of such expert and intelligent systems.

1. Introduction

Augmented reality (AR) consists in real time addition of virtual information that is coherently positioned with respect to a real environment. AR technology is used in the automotive industry for various applications (Nee, Ong, Chryssolouris, & Mourtzis, 2012), such as service training and assistance (Stanimirovic et al., 2014). In order to fulfill its goal, an AR system needs to continuously perform real time estimation of its position and orientation in 3D relative to the real world, which is a task known as tracking. A common way to accomplish this is by detecting planar fiducial markers placed around the environment using a video camera (Kato & Billinghurst, 1999). Nevertheless, in an automotive context, such markers can be considered intrusive, so it is more suitable to rely on natural features present in the real world. Such approach brings many challenges, such as dealing with large scale scenarios, objects small parts, variable illumination conditions, and materials with low texturedness, reflective and transparent properties, just to cite a few.

This work presents a complete markerless tracking solution suited to the development of AR applications for the automotive industry. The system adopts a model based tracking by detection that relies on keypoint features and covers the phases of model generation, calibration and tracking itself.

The contributions of this paper are:

1. A pipeline for markerless tracking designed for the creation of AR systems that target the automotive sector, but are also suitable for other application scenarios;

* Corresponding author at: Departamento de Estatística e Informática (DEINFO), Universidade Federal Rural de Pernambuco (UFRPE), Rua Dom Manoel de Medeiros, s/n - Dois Irmãos, 52171-900 - Recife, PE, Brazil. .

E-mail addresses: jpsml@cin.ufpe.br, joao.mlima@ufrpe.br (J. Paulo Lima), rar3@cin.ufpe.br (R. Roberto), fpms@cin.ufpe.br (F. Simões), mwsa@cin.ufpe.br (M. Almeida), lsf@cin.ufpe.br (L. Figueiredo), jmxnt@cin.ufpe.br (J. Marcelo Teixeira), vt@cin.ufpe.br (V. Teichrieb).

http://dx.doi.org/10.1016/j.eswa.2017.03.060
0957-4174/

2. A semi-automatic method for reconstruction of scenes with undesirable parts;
3. A tracking quality checking method based on inlier count and reprojection error metrics;
4. Evaluations on automotive sector scenarios proposed by the Volkswagen/ISMAR Tracking Challenge 2014, where performance and tracking quality of the proposed system are measured.

This paper is organized as follows. Section 2 presents existing concepts and works regarding tracking, AR and the automotive industry. Section 3 describes the proposed markerless tracking system. Section 4 gives details about the tracking challenge that was used to evaluate the system. Section 5 presents and discusses the results obtained in the evaluations performed. Section 6 draws conclusions and points out future work.

2. Background and context

The following subsections present major relevant concepts and existing works about tracking, AR and the automotive application domain.

2.1. Tracking and AR

In computer vision, tracking is defined as retrieving the part of the image that contains the target object (Yilmaz, Javed, & Shah, 2006) (Smeulders et al., 2014) . Given the 3D model of the targeted object, the tracking task is analogous to retrieving the camera pose. It establishes a direct relation between the camera and the object by acquiring the knowledge of where the camera is and to which direction it is pointing at. Using this relation, it is possible not only to identify in which part of the image the object is located but also the distance from it to the camera and its relative orientation.

However, in order to use a tracking solution in an AR application (Marchand, Uchiyama, & Spindler, 2016), some additional requirements arise. At first, the camera pose must be retrieved in real time so the user can perceive the augmentation and interact with it normally. Moreover, it is required precision so the augmented content may not be misplaced on the scene, providing wrong or ambiguous information to the user. Additionally, in several scenarios, it is desirable to perform the tracking steps without the need to add markers on the scene. Markerless trackers are likely to expand the applicability range being less intrusive and usually requiring minimum or zero setup effort of the final user.

2.2. Automotive application domain

Considering the automotive domain of application, tracking results can be used for several purposes, in scenarios when the user is both inside and outside the vehicle. The following discussed scenarios will target mobile platforms, varying from tablets and smartphones to see-through head-mounted displays (HMDs). Some of the described scenarios are still in the conceptual phase while others are already implemented and in use.

The application scenarios here addressed are directly related to the technical challenging scenarios targeted by our tracking solution. These challenges are: track the car engine; track tire and wheel; track the car interior, including panel and dashboard; track car tools such as the jack and the lug wrench; and track the entire car from an outside point of view. Furthermore, in Section 4 each one of the three case studies explores the tracking technical challenges linked to each one of the following application areas.

2.2.1. Training and maintenance solutions

The training and maintenance tasks have been tackled in several application domains by AR solutions (Makris, Pintzos, Rentzos, & Chryssolouris, 2013). On these tasks, the mechanic is usually aided by precisely located augmented content with instructions, highlighted parts or preview animations illustrating what needs to be done in the current step. Regarding the automotive domain, examples arise targeting different parts of the vehicle, usually focusing on the engine itself or on its specific parts. However, other parts may be targeted, such as the car hood, wheel (Stanimirovic et al., 2014) or even the door (Reiners, Stricker, Klinker, & Mller, 1998). Fig. 1 shows some examples. Some of these examples rely on markers to explore the goal functionality as a research target (Henderson & Feiner, 2011; 2007; Lee & Rhee, 2008). However, in addition to the previously discussed issues related to marker based tracking systems, when dealing with both training and maintenance tasks, it is likely to have markers occluding part of the workspace. Occlusion can also happen when used tools and user hands block the line of sight from the camera viewpoint to the markers, potentially causing tracking failures. Thus, there is a clear concern by researchers and industry to provide markerless tracking solutions for these scenarios (Platonov, Heibel, Meier, & Grollmann, 2006), (Stanimirovic et al., 2014).

2.2.2. Flat tire replacement

Particularly, the task of changing the car tire is often performed by drivers. The flat tire event is not a controlled one. It occurs in unplanned places and in some situations it may cost time to get additional mechanical assistance to the driver. Thus, systems that provide in-place augmented assistance are useful to help the driver along this task. Two tracking challenges arise in this case: as shown in Fig. 1, it is required a system to track the car wheel/tire (Lee & Rhee, 2008) and it is also useful for the system to be able to track car tools. The tools will be used in the process, and the system should help the driver to operate them correctly and efficiently. Fig. 2 illustrates examples of systems tracking medium-sized objects with complex shapes, low texture and metallic surfaces. The challenges relative to tracking these objects can be related to tracking common car tools, such as the jack and the lug wrench, which are used in the tire replacement process.

2.2.3. Driver vision helper systems

AR systems are able to help the driver while inside the car in several different ways, from facilitating the use of the car features (e.g. placing annotations and additional information in the car panel or dashboard), to helping the driver on passing and lane change tasks. Over the years car models are incorporating new features, adding buttons to the steering wheel, increasing the panel interactive region and providing new information on the dashboard. In these cases, augmented content can be used to help the user in the task of finding a specific functionality inside the car. The complexity of such tasks can be calculated based on the work from Rentzos, Vourtsis, Mavrikios, and Chryssolouris (2014). Fig. 3 shows on the left the use of a spatial AR system that aims to prototype different panels and dashboards configurations by projecting content in a previously calibrated environment (Porter, Marner, Smith, Zucco, & Thomas, 2010). Similar augmentations could provide dynamic content over the panel to expose its features for the user by exploiting real time tracking on the car interior. On the right side, Fig. 3 shows a system that provides a virtual environment which simulates the car interior, where an HMD is worn by users in order to achieve immersion (Salzmann & Froehlich, 2008). In the AR context, HMDs may be replaced by see-through glasses and, by tracking the car interior from the user viewpoint, all needed cues for specific panel procedures (turning on the heat, changing music volume, locking or unlocking doors, and so on) can be rendered at the user glance.

Fig. 1. AR applications for training and maintenance for the automotive application domain, targeting the car engine (Lee & Rhee, 2008) (top row), the car hood and wheel (Stanimirovic et al., 2014) (bottom row).

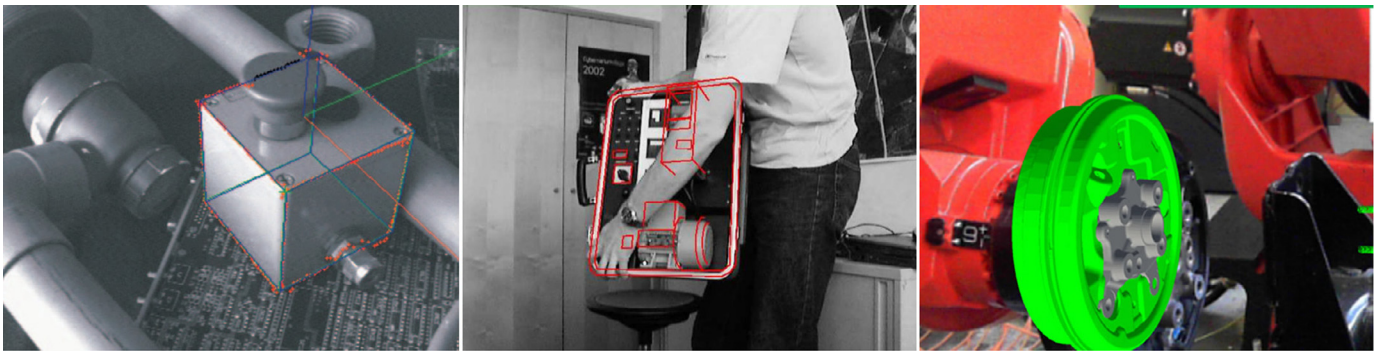

Fig. 2. Examples of tracking solutions handling complex shaped objects. At the left a part of a hydraulic system being tracked in real time (Comport et al., 2006), at the center an object being tracked while it is being manipulated (Wuest, Vial, & Strieker, 2005), and at the right a digital component superimposed over real objects (Makris et al., 2016).

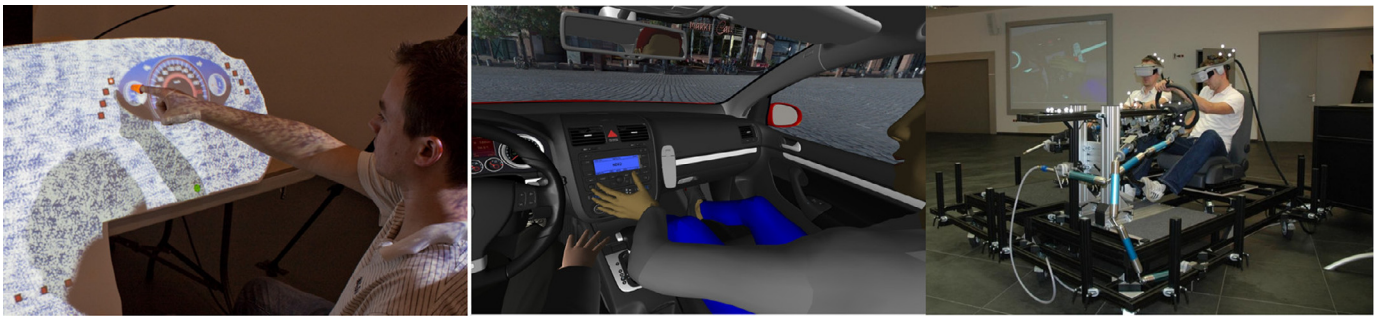

Fig. 3. Example of spatial augmentation of a car panel on the left (Porter et al., 2010). On the right a Virtual Reality system that tracks the user viewpoint to correctly render the content on the HMD (Salzmann & Froehlich, 2008).

Fig. 4. Example of inspection task using AR (Georgel et al., 2007).

2.2.4. Car bodywork inspection, customization and traffic control

Another set of applications arises by looking at the car as a single piece. As illustrated in Fig. 4, the inspection task is well-known as a target for AR applications because by aligning real and augmented content the discrepancy check is facilitated (Georgel et al., 2007). Given the 3D model of the targeted car and a precise tracking system, it is possible to identify failures on the car bodywork during an inspection procedure, for example. Moreover, by tracking the car exterior as a whole, AR systems can enable users to customize its appearance, add external accessories and change its color. At last, real time tracking of car models can be used by expert systems for purposes related to traffic control, in a search for a missing car and security when analyzing vehicles trajectories and preventing car crashes (Koller, Weber, & Malik, 1994).

3. Markerless tracking system

Natural feature tracking techniques for AR need 3D knowledge about the object, which is referred to as a model of the object. This model can be encoded in different ways depending on the method's requirements, such as computer-aided design (CAD), 3D point cloud and plane segments. The tracking system described in this work can be classified as a model based one, since it makes use of a previously obtained model of the target object, in this case a car. Model based systems are able to handle scenarios where the object and/or the camera move with respect to each other. The proposed system can also be classified as a detection system, since it is able to calculate the object pose without any previous estimate, allowing automatic initialization and recovery from failures. Fig. 5 depicts the input and output of all system's modules, which are described in the following subsections.

3.1. Model generation

An overview of the submodules that belong to the model generator module is illustrated in Fig. 6. In order to be able to reconstruct the entire car, which has low textured materials and small parts, an RGB-D sensor was employed. It provides in real time, besides a color image (RGB channels) of the scene, another image in which each pixel value corresponds to the distance between the scene objects and the camera (Fig. 7) named depth image (D channel).

First, an RGB-D sequence of the car areas to be tracked is captured in a single take. This is done in order to have all the reconstructed parts of the car in the same coordinate system. Then, the scene is reconstructed from the depth data of the sequence using KinectFusion (Newcombe et al., 2011). KinectFusion performs real time reconstruction when a fixed volume of the 3D space with specific dimensions is reconstructed (in this case, a volume with $512 \times 384 \times 512$ voxels). Therefore, it was adopted since it is desirable that the reconstruction phase does not take too much time. In addition, the reconstruction resolution (voxels per meter) is decreased for covering the entire car. Fig. 8 illustrates the result of the reconstruction, which is a colored 3D point cloud of the car.

After that, some keyframes are selected for generating the model to be used in the tracking phase. A keyframe is basically an image of the car with a known pose, making it possible to estimate the corresponding 3D coordinates of a given 2D feature extracted from it. In the proposed system, candidate keyframes are selected by considering a fixed frame interval. A candidate keyframe is selected at every n frames and tests were performed to determine the ideal interval. Then the user manually browses the candidate keyframes and chooses which ones will be retained. While selecting the keyframes, the user is able to group the ones that cover a specific part of the car (e.g. engine, fender, interior, trunk). It is also possible to refine the model by manually removing undesired areas of the keyframes (e.g. floor, specular parts). This is done using a brush tool and the user has to paint the areas in the color image that should be removed from the 3D point cloud, as shown in Fig. 9.

Once the keyframes are selected and refined, their grayscale images are normalized to zero mean and unit variance in order to cope better with illumination changes. Then 2D keypoints with associated binary descriptors are extracted from keyframes normalized grayscale images using the ORB detector (Rublee, Rabaud, Konolige, & Bradski, 2011), as depicted in Fig. 10. The corresponding 3D points in the cloud for each 2D keypoint are then obtained, which will be referred from now on as 3D keypoints. The final model for each part of the car is generated in an incremental way. First, the 3D keypoints of the first keyframe of the part are added to the model. Then, the keypoints from the next keyframe are matched to the current model keypoints using a nearest neighbor search based on the Hamming distance between their binary descriptors (Rublee et al., 2011). A heuristic is applied to reject spurious matches, in which a correspondence is discarded if the ratio between the distances of the closest and the second-closest neighbor is less than a threshold (Lowe, 2004). In the proposed system, this threshold was set to 0.7. Only the keypoints that are not matched to the current model are added to it, in order to avoid the presence of repeated features. The same procedure is adopted for the remaining keyframes. The model stores 3D keypoints and also the descriptors of their corresponding 2D keypoints.

3.2. Calibration

Once the model generator creates the 3D model of the car in its particular 3D coordinate system, the calibrator module is responsible for transforming the recovered model from its coordinate system to another one. This transform allows the application to be flexible when addressing the model, by using real world measurements, a pre-acquired CAD model of some equipment or a factory floor coordinate system just to cite a few. It is worth noting that the calibration step is optional for tracking a given scene, being only required if it is desired to augment the world with virtual objects placed at 3D coordinates relative to a real world coordinate system.

Since the recovered model preserves the correct shape of the car without distortions, which means to have a Euclidean reconstruction, the calibrator module can compute a similarity transform that gets from one Euclidean model to another Euclidean model based on 3D points correspondences. This similarity trans-

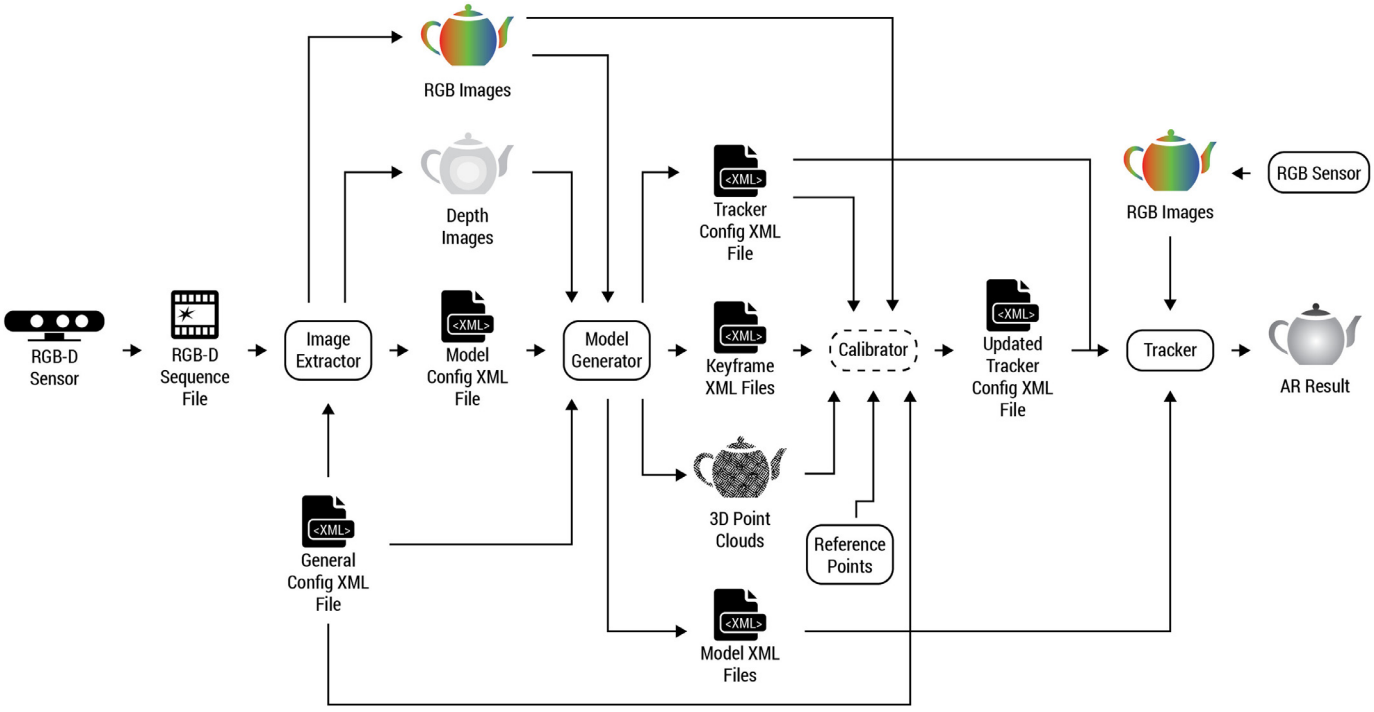

Fig. 5. System pipeline overview.

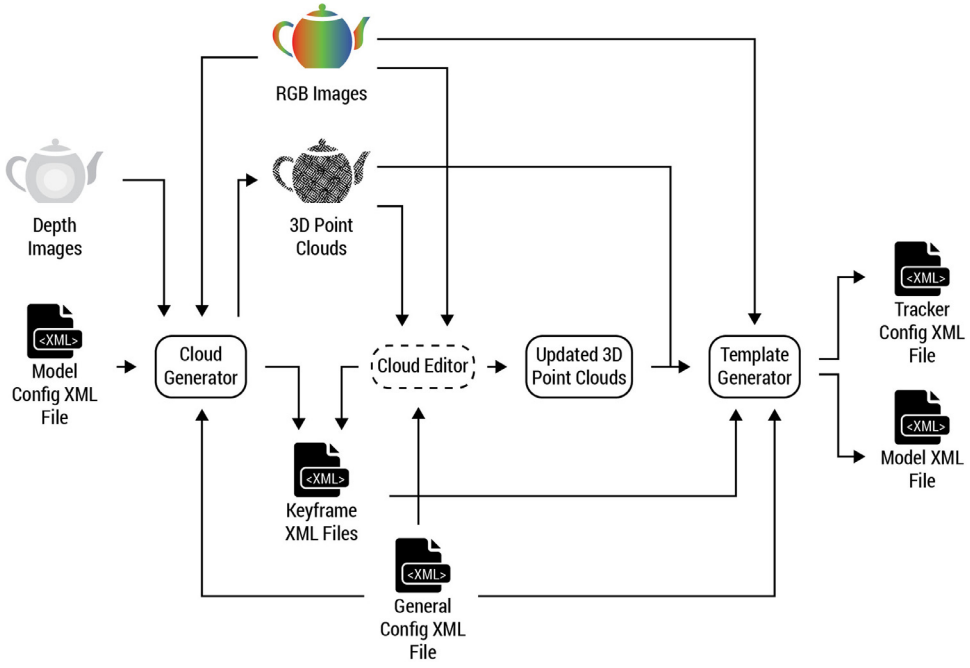

Fig. 6. Model generator overview.

form is defined as a 4x4 matrix in homogeneous coordinates that comprises a rotation, a translation and also a scale transform. In this system, the calibrator module employs a closed form absolute orientation technique (Horn, 1987) to estimate the similarity matrix. It uses at least three points correspondences between the models and has the advantage of being easily scalable, fast to compute and precise.

The calibrator module receives as input the generated 3D model from the previous one and the reference points of the new model as seen in Fig. 5. Based on these coordinates, the user is required to manually select the 3D points from the generated model that correspond to the reference points. In order to correctly match these points, the user needs to select each corresponding point from one valid keyframe in the same sequence of the reference points. The system allows the user to iterate through the keyframes to choose the one where the point is better exposed and also to magnify the image to pick the point precisely as shown in Fig. 11.

3.3. Tracking

As seen in Fig. 5 and explained in the previous subsections, the model generator and calibrator modules create all the files re-

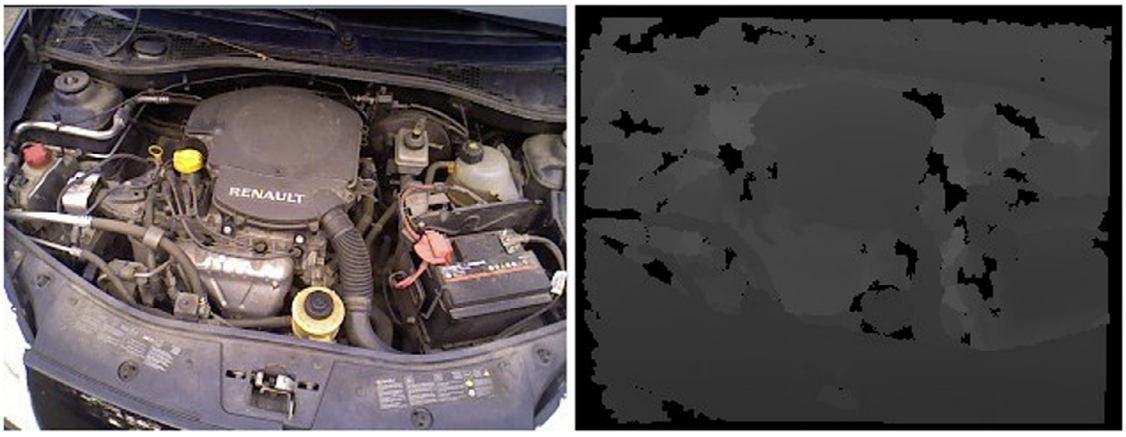

Fig. 7. Color (left) and depth (right) images of a car engine provided by an RGB-D sensor.

Fig. 8. 3D point cloud of a car computed from RGB-D data.

Fig. 9. Manual model refinement tool for removing undesired areas, which are painted in red. For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.

Fig. 10. ORB keypoints extracted from an image of a car engine.

Fig. 11. Calibration tool for selecting reference points in the keyframes.

quired for the online tracking phase. Additionally to the model and calibration data, the 3D points to be tracked in world coordinates are also an input to the tracker module.

The tracker's first step is to extract and describe 2D keypoints from the images captured by the RGB camera using the ORB detector. After that, the system matches the current image keypoints with the set of keypoints from the corresponding group that covers the captured part of the car. Similar to the model generator module, the matcher uses a nearest neighbor search based on the Hamming distance between their binary descriptors. Also, the

Table 1
Summary of quality measurements and color meanings.

Criteria	Indicator Color	Meaning
reprojection error <3 AND number of inliers >15	Green	There is a high possibility that the position indicated is accurate
3 <reprojection error <5 OR 8 <number of inliers <15	Yellow	There is a low possibility that the position indicated is accurate
reprojection error >5 OR number of inliers <8	Red	There is a high possibility that the position indicated is inaccurate

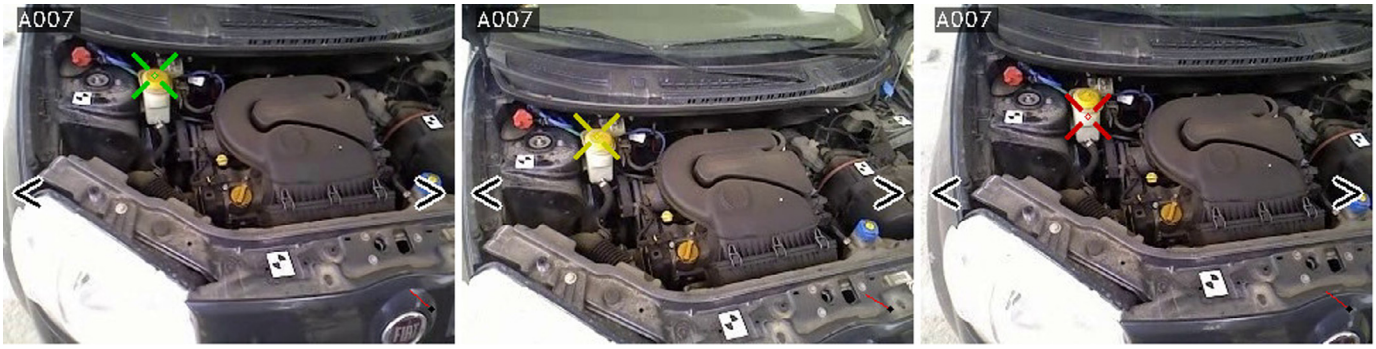

Fig. 12. Examples of tracking quality checking when the display indicator color is green (left), yellow (center) and red (right). For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.

same heuristic that discards a correspondence if the ratio between the distances of the closest and the second-closest neighbor is less than a threshold is applied to reject spurious matches. The difference is that this threshold starts with 0.7 but the user is able to change this value in real time to enhance results. Given the set of matches, it is possible to estimate the current frame pose from the 2D-3D correspondences using EPnP estimator (Moreno-Noguer, Lepetit, & Fua, 2007) combined with the RANSAC algorithm (Fischler & Bolles, 1981) for outlier removal.

In order to measure the tracking quality, the system calculates the average reprojection error from the 3D keypoints used to estimate the pose. This information, along with the number of inliers calculated by EPnP, is used to give visual feedback to the user regarding the tracking quality. Experimentally, it was possible to see that there is no guarantee of the tracking quality when there are less than eight inliers or the reprojection error is higher than five pixels. Thus, the display indicator that leads to the 3D points to be tracked at the car becomes red. Additionally, the system achieved good results when there are more than 15 inliers and the reprojection error is below three pixels and the display indicator becomes green. Intermediate values are shown in yellow. This quality measurement is summarized in Table 1 and illustrated in Fig. 12.

4. Case study

Since 2008, the International Symposium on Mixed and Augmented Reality (ISMAR) stimulates research in the area of tracking by promoting a contest known as “ISMAR Tracking Competition”, in which tracking systems are applied to real industry problems. Volkswagen, one of the main vehicle manufacturers in the world, since 2013 sponsors its own competition, called “Volkswagen Tracking Challenge”. In 2014 this contest was included as part of the ISMAR conference, replacing the customary competition.

Thereby, the main industrial problems to be addressed by the challenge that year were related to the automotive domain, mainly regarding markerless tracking techniques, with the main focus on those used for accurate tracking of vehicle components, which could help to increase even more the research and application of AR in this domain.

The challenge contained scenarios in which our ever developing tracking techniques could be tested to the maximum of their capabilities. The main purpose of each scenario was to be able to detect and keep tracking certain reference points in dynamic and sometimes noisy environments.

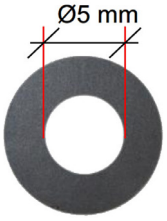

Fig. 13. Example of reference/challenge point uncoded circular marker (AG, 2014).

Each scenario was divided into a preparation and a competition phase that contained some tasks - organized by difficulty - to be performed. Rating points were given by the jury of each task, which was composed of members from the Technical University of Munich and employees of Qualcomm Incorporated, ART Advanced Realtime Tracking GmbH and Volkswagen AG.

In the next subsections we describe these scenarios, giving details about input data and elements used in the process, the preparation and competition phases and their respective tasks as well.

In order to initialize the system, mainly in the preparation phase, some information was given to the participants to be used as input to the system. These data (reference points coordinates) had the purpose of allowing registration to the respective scenario coordinate system, even though in scenarios 1 and 2 this initial registration was also possible through the use of CAD data (provided as well). It is important to remember that these reference points were removed from the scene after the preparation phase.

These points are marked using uncoded circular markers of known dimensions (as illustrated in Fig. 13). For each circular marker present in the scene, the corresponding 3D coordinate was provided as well. Each marker had a four element ID in the format “Rxxx”, where the “xxx” is the number of the marker (padded with zeros from the left), e.g. “R040”. It should be noted that the markers are only used for calibration purposes and are not taken into account for model generation and tracking procedures, which are fully based on natural features.

The format of the input file was also described as being a simple ASCII formatted file. And for the reference and challenge points (the points which need to be tracked and identified by the partic-

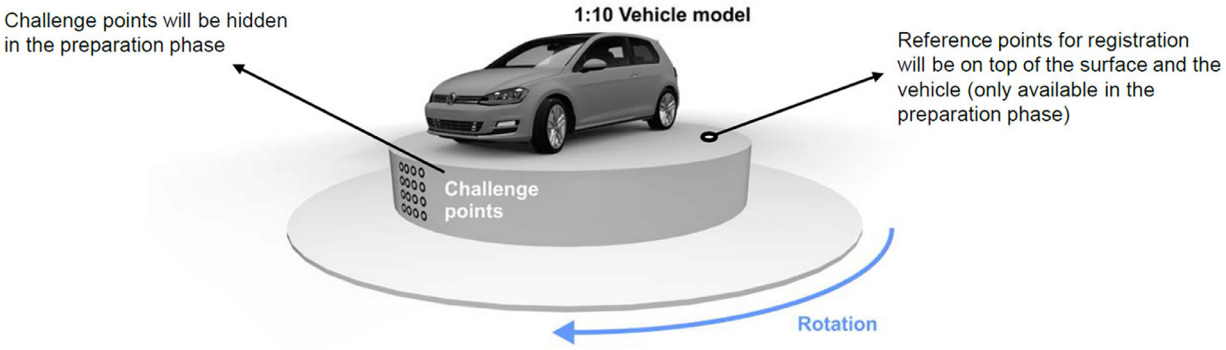

Fig. 14. Front view of scenario 1 setup (AG, 2014).

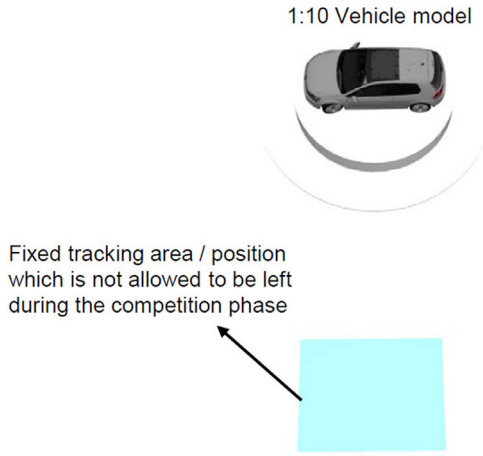

Fig. 15. Top view of scenario 1 setup (AG, 2014). For interpretation of the references to color in the text, the reader is referred to the web version of this article.

ipants), the 3D coordinates followed their ID. For instance, a challenge point named "A001" was described in this way in a line of the input file: **A001;123.1;0.6;-120.2**. This point description was terminated by a Unix EOL character (0x0A).

4.1. Scenario 1

The first scenario of the tracking competition comprised tracking a rotating vehicle. The corresponding setup was a 1:10 vehicle miniature placed over a rotating platform, as shown in Fig. 14. Both rotation speed and direction could be changed by the judges.

The task given was to exactly locate the 3D coordinates of the rotating vehicle model by determining the corresponding challenge points or parts. In a second moment, virtual data (the car 3D model) should be correctly superimposed onto the real vehicle model while it rotated.

Scenario 1 preparation can be described as follows. At first, from a given tracking area, the competitors had to register to the local vehicle coordinate system using the circular markers or known 3D data (vehicle 3D model). In the competition phase, points had to be identified using predefined 3D coordinates. Afterwards, the rotating vehicle model had to be overlaid with the given 3D data as accurate as possible.

The three main tasks of scenario 1 were: track the rotating car model from a fixed position, highlighted in blue in Fig. 15; visualize and identify 3D challenge points; and overlay the car model with a 3D structure.

The competition provided 3D data parts of the car model in millimeters in different formats (OBJ, VRML and STL). The same 3D model should be overlaid over the real vehicle. Also, reference points were given with their corresponding 3D coordinates.

The inherent challenge in this scenario was to track, from a fixed position, moving objects with variable speed and to identify 3 challenge points per task, totaling 9 points at all. Table 2 describes the 4 tasks in detail.

4.2. Scenario 2

The second scenario involved capturing and tracking of different parts of a real Volkswagen Golf™. There were four sequential tasks which involved acquiring the exact determination of following parts of the vehicle defined by 3D coordinates (illustrated by Fig. 16).

In the preparation phase, the competitor was allowed to capture the external appearance of the complete vehicle and register with the help of the circular markers in the engine part, which had its 3D coordinates provided. Besides that, the participant could use 3D data of parts of the exterior and interior in the local coordinate system of the vehicle. It is worth noting that, specifically in this scenario, the scene could be changed between the preparation and the competition phases by, for instance, adding or removing light.

The competition phase consisted of the proper execution of tracking vehicle's interior and exterior following some constraints, involving the definition of each area to be tracked as a separate task in a predefined order and using restricted tracking areas. In addition, tracking was not allowed while moving from one area to the next one. Fig. 17 illustrates these constraints.

During the tracking phase using the system, the competitor should identify some corresponding parts on the vehicle by directly pointing at a predefined element with the finger.

The challenge organizers provided 3D data parts corresponding to the car exterior in millimeters in different formats (OBJ, VRML and STL) and the reference points on the car engine with their 3D coordinates.

The first task - tracking the engine part - main purpose was that the system used given information (reference points) as basis for the tracking in this area and in the rest of the car, as these points would help to define the complete coordinate system of it. The main tracking challenge here was to find hidden or difficult to see challenge points in limited tracking areas.

The second task - tracking the interior from driver's seat - was to continue the tracking using an extrapolation of the initialization provided by the engine compartment. Therefore, for this area there were no reference points available. For this task, there was also the possibility of changes in the scene before the competition phase.

The third task - tracking of the trunk - required the capability of tracking from a bright into a dark environment, containing a sparse amount of features.

Table 2
Overview of tasks regarding scenario 1.

Task	Initialization Rotation	Challenge Rotation	Display	Difficulty	Points	Max Score
1	none	constant	3D points	2	3	6
2	constant	constant	3D points	3	3	9
3	constant	variable	3D points	4	3	12
4	none	variable	3D data	Score given by jury		12

Fig. 16. Illustration of the vehicle parts to be tracked in scenario 2 (AG, 2014).

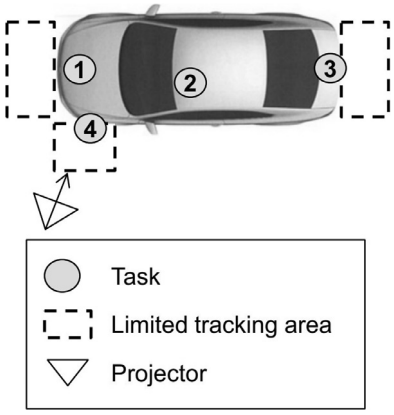

Fig. 17. Sequence of tasks and tracking area constraints. The projector's purpose is to possibly add disturbances by light in the fender area (AG, 2014).

The last task - tracking the fender - main challenge was the ability to track from a short distance with a low number of features. As shown in Fig. 17, there was a projector which could add disturbances caused by artificial lights, turned on right before the competition phase.

Table 3 depicts more details about each task.

Table 3
Overview of tasks regarding scenario 2.

Task	Area	Area size	Difficulty	Points	Max Score
1	Engine	$< 2\ m^2$	2	4	8
2	Interior	Complete interior	3	4	12
3	Trunk	$< 2\ m^2$	4	3	12
4	Fender	$< 1\ m^2$	3	3	9

4.3. Scenario 3

The third scenario of the tracking competition comprised tracking objects with high accuracy. The corresponding setup was a table with some objects placed over it, as shown in Fig. 18.

The given task was to use reference points along with objects information to learn the 3D coordinates for the entire scene. In a second moment, we should accurately place markers on the given 3D coordinates.

Unlike scenarios 1 and 2, in this scenario the preparation and competition phase directly flowed into each other. During the preparation phase, the contestants were allowed to place their own markers and features into a specified area in the center of the table. These markers and features had then to be registered to the local coordinate system which was defined by the circular markers. The exact 3D coordinates of the reference points were provided.

The challenge inherent in this scenario focused on both speed and precision. The preparation phase took about 30 min, while the

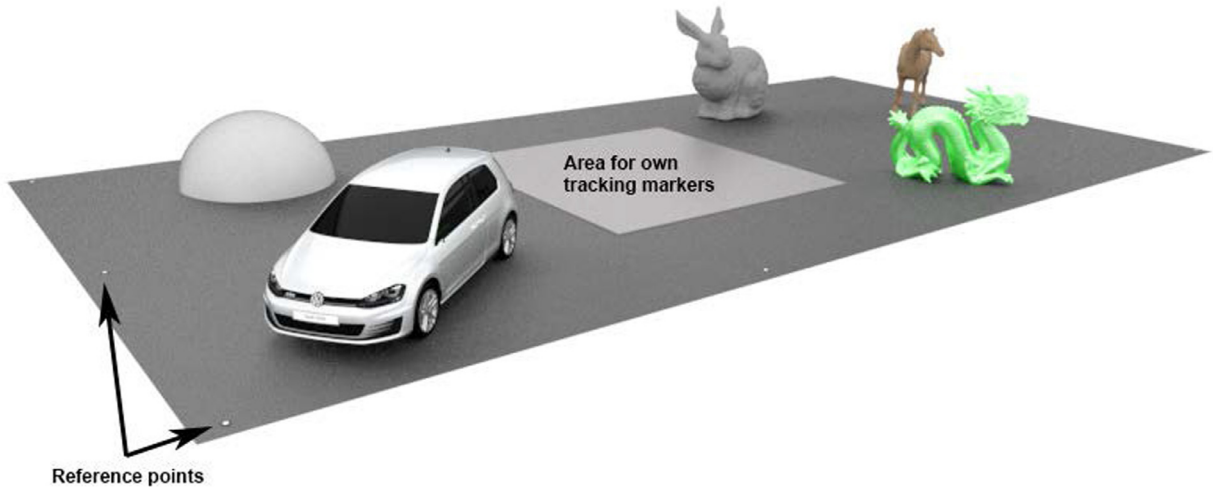

Fig. 18. Scenario 3 setup (AG, 2014).

competition phase lasted for at most 15 min. After a photogrammetric measurement of the placed markers, the mean of the distances to the correct coordinates were calculated to find the most accurate results. 1–3 scores were given based on speed for overall placement of all 4 markers. 4–6 scores were given based on the precision of overall placement of all 4 markers. In sequence, scores were summed up and the winner was the participant with the highest score.

5. Results and discussion

The tracking system was written in C++ and executed on the Microsoft Windows 8.1 operating system. The following libraries were used in the implementation of the system: OpenCV,[1] Point Cloud Library (PCL),[2] OpenNI,[3] Microsoft Kinect SDK and Developer Toolkit[4] and videoInput.[5] The hardware used in the tests was an Asus Xtion PRO LIVE and a Microsoft Surface Pro-1 tablet with Intel Core i5-3317U @ 1.70 GHz processor, 4GB RAM and an Intel HD Graphics 4000 display adapter.

All the sequences were captured with a resolution of 320 × 240 pixels. For scenarios 1 and 3, it was used a reconstruction resolution of 256 voxels per meter. For scenario 2, the resolution was decreased to 64 voxels per meter in order to allow reconstructing the car completely. An interval of 20 frames was adopted for generating keyframes from the captured sequences. In scenario 1, an elliptical mask was applied to all keyframes in order to retain only the area around the car miniature. This was done to allow KinectFusion to reconstruct the scene correctly, since it would interpret that the camera was moving around the car. Regarding keypoint extraction, the ORB feature detector was configured to return the best 500 features.

Tracker performance was evaluated using a model with 52,588 3D keypoints. Descriptors nearest neighbor search was performed using locality-sensitive hashing (LSH) (Lv, Josephson, Wang, Charikar, & Li, 2007). Table 4 presents the mean and standard deviation of time measurements for each step of the tracking pipeline. It can be noted that mean frame rate of the tracker is 4.83 (0.29) fps, which allows an interactive AR experience. The bottleneck is the keypoint matching procedure, which takes more than 90% of all processing time.

Table 4
Mean and standard deviation of the time required by each step of the tracker.

	Mean (SD) time (ms)
Keypoint detection	7.52 (1.33)
Keypoint matching	188.46 (12.41)
Pose estimation	11.62 (2.76)
Total	**207.60 (12.71)**

Fig. 19 illustrates some results obtained in each scenario of the tracking competition. The colored dots refer to the projection of the 3D point cloud model of the scene using the pose computed by the tracker. In scenario 1, the developed system scored 23 out of 27 in the challenge points identification tasks. However, there were some misregistrations while superimposing the virtual car 3D model onto the real vehicle, mainly due to the fact that our system performs tracking by detection without taking into account temporal information. Using a recursive tracking approach for this task would probably provide a better experience to the users, where the pose of the previous frame is used as an estimate for the current frame pose. Such kind of method is often faster, more accurate and more robust to noise, but is not able to perform (re)initialization. In scenario 2, the system experienced some difficulties related to environment lighting changes and lack of extracted features in some situations. In scenario 3, the measured accuracy of markers placement was 12.20 mm and the time needed to place the markers was 7 min 08 s.

In addition to the results obtained in the competition, the competition environment was recreated in order to acquire a new dataset, which allowed performing additional evaluations of the tracking system. The reference points coordinates were gathered using a Leica FlexLine TS06plus manual total station with an angular accuracy of 2" and a linear accuracy of 1.5 mm+2 ppm. Since a miniature of the car used in the tests was not available, only scenarios 2 and 3 were prepared for the evaluation. For scenario 2, the coordinates of 8 points in the engine, 6 in the interior, 3 in the trunk and 3 in the left fender were measured to serve as challenge points. For scenario 3, the coordinates of 10 arbitrary locations on the table were obtained. Competitors were asked to mark the location of the challenge point with a pen and the distance between marked and correct position was measured using the total station. Each competitor had to mark 5 challenge points

[1] http://opencv.org/.
[2] http://pointclouds.org/.
[3] http://structure.io/openni.
[4] https://dev.windows.com/en-us/kinect.
[5] http://www.muonics.net/school/spring05/videoInput/.

Fig. 19. Tracking results in the actual competition: scenario 1 (first row), scenario 2 (second, third, fourth and fifth rows) and scenario 3 (sixth row).

previously selected from the 10 existing ones. Fig. 20 shows some results obtained with the acquired dataset, where the display indicator highlights the challenge point to be identified by the user. It should be noted that in the scenario 3 screenshots the indicated positions are reference points in one of the corners of the marker grids used.

The evaluation results relative to recreated scenario 2 are summarized in Table 5. Two different conditions were tested: single reconstruction (SR), where the entire car was reconstructed in the same coordinate system; and multiple reconstructions (MR), where each car part was reconstructed separately and they were not aligned with each other. Six competitors aged from 28 to 31 participated in this evaluation. One half of the competitors consists

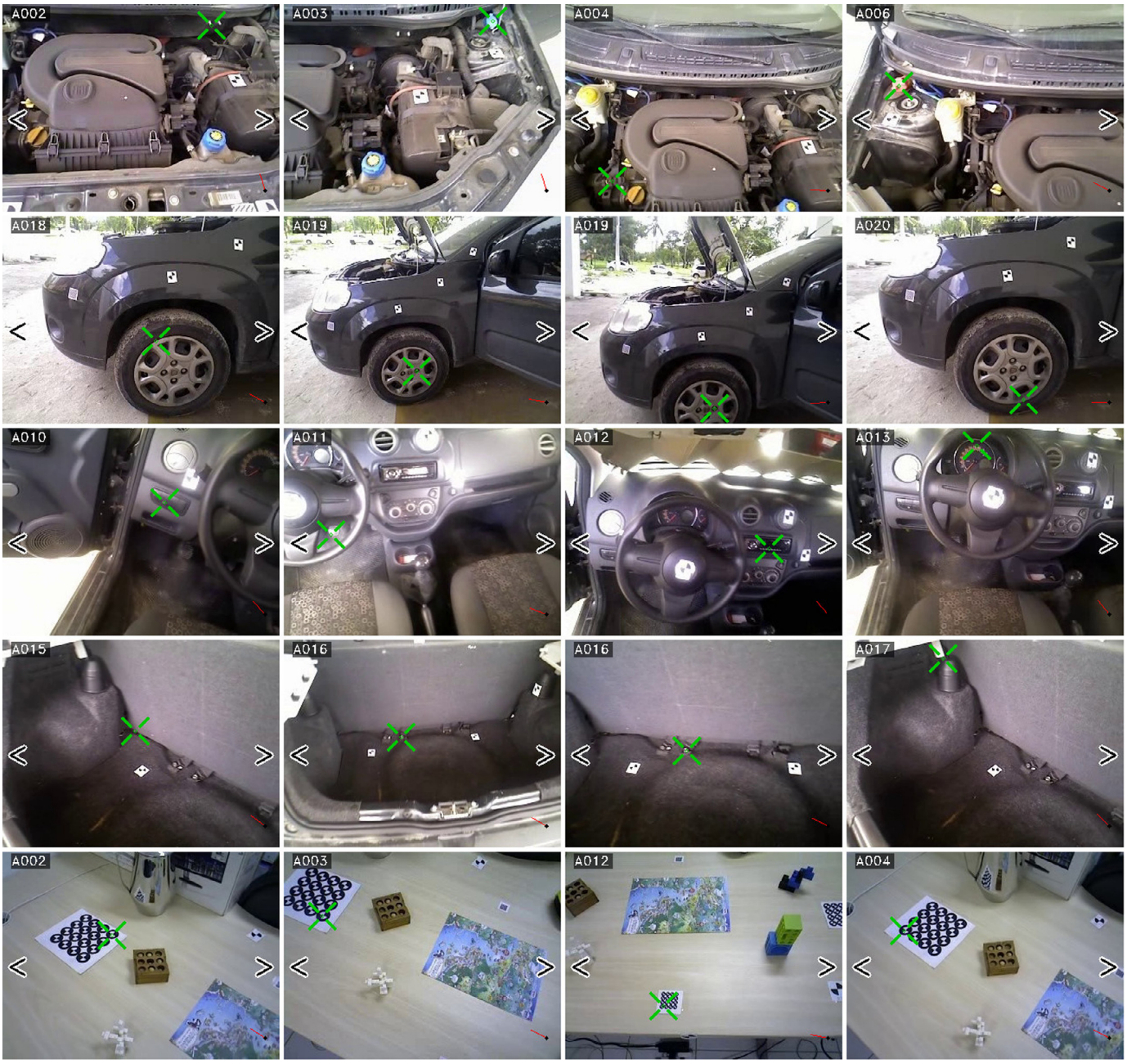

Fig. 20. Tracking results in the recreated competition: scenario 2 (first, second, third and fourth rows) and scenario 3 (fifth row).

of members of the system development team and the other half was composed of regular users that had no contact with the application before. It can be noted in Fig. 21 that competitors obtained significantly better results when using the MR configuration. Moreover, they achieved the result in less time, as seen in Fig. 22. This can be explained by the fact that in the SR condition the entire car is reconstructed using a single take, which demands the use of a lower reconstruction resolution and also causes error accumulation. In the MR configuration, since smaller volumes are reconstructed one at a time, a higher resolution can be used and less error is accumulated. The competitors were not able to identify many points in the vehicle interior, due to the influence of environment lighting and to the lack of discriminative keypoints. There was not a significant difference between the number of correctly identified points by developers and regular users, but Fig. 22 shows that in general, developers took less time than regular users to accomplish the task. These results suggest that when the proposed system is used in a part localization task such as scenario 2, the level of expertise may affect execution time but does not have too much impact on execution correctness.

Table 6 depicts the results obtained with recreated scenario 3. This test also involved six competitors, aged from 27 to 31, and they were also equally divided into developers and regular users groups. Most of the measured accuracy values were similar to the value reported in scenario 3 of the actual competition. Both regular users #2 and #3 presented a large misplacement in one of their challenge points, which harmed their accuracy results, as well as of this entire group, as seen in Fig. 23. Fig. 24 shows that the time difference between developers and regular users was not significant. The results of this evaluation suggest that expert users are more likely to perform better in high accuracy tracking tasks such as scenario 3.

Table 5
Number of correctly identified challenge points for each car part and time spent by each competitor in recreated scenario 2.

	User #1		User #2		User #3		Dev #1		Dev #2		Dev #3	
	SR	MR	SR	MR	SR	MR	SR	MR	SR	MR	SR	MR
Engine (8 points)	6	6	6	6	7	7	6	7	6	6	6	6
Interior (6 points)	1	0	0	2	1	0	0	0	0	0	0	0
Trunk (3 points)	0	3	0	1	0	1	0	3	0	1	0	1
Fender (3 points)	0	2	1	3	1	3	1	3	1	1	2	3
Total	**7**	**11**	**7**	**12**	**9**	**11**	**7**	**13**	**7**	**8**	**8**	**10**
Time (min)	24	5	33	30	22	16	19	11	13	6	24	10

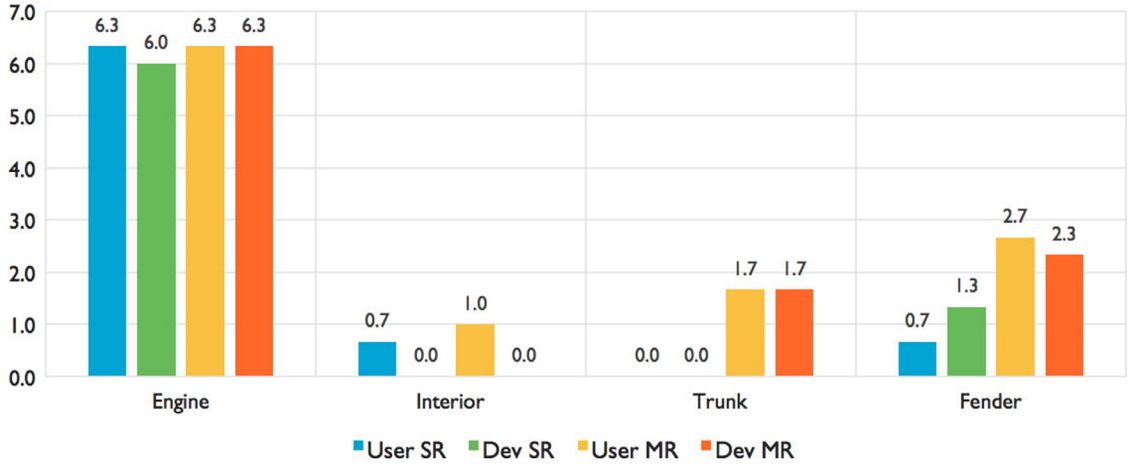

Fig. 21. Average number of correctly identified challenge points for each car part by the two groups of competitors in recreated scenario 2.

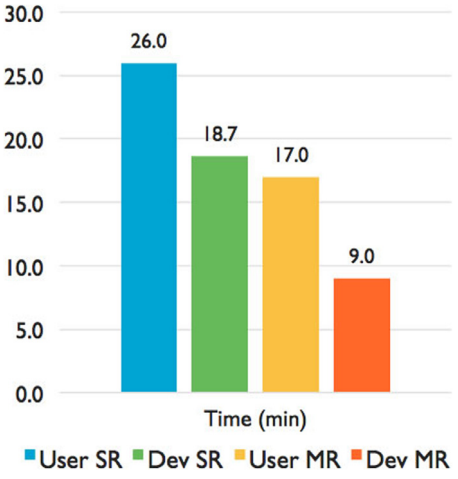

Fig. 22. Average time in minutes spent to identify challenge points by the two groups of competitors in recreated scenario 2.

Table 6
Mean and standard deviation of tracking accuracy and time spent by each competitor in recreated scenario 3.

	Mean (SD) accuracy (mm)	Time (min)
User #1	12.30 (6.29)	11
User #2	93.82 (177.51)	7
User #3	44.29 (50.42)	9
Dev #1	13.95 (15.81)	8
Dev #2	13.96 (10.55)	11
Dev #3	8.02 (4.13)	9

While developing and testing the proposed pipeline, integration issues were handled and pitfalls were overcome. As a result of the experience of applying theoretical concepts on a practical end-to-end pipeline for 3D models reconstruction, tracking and augmentation, the following lessons learned and insights can be presented:

- The use of keypoint features (such as ORB), are usually applied on highly textured environments (e.g. crowded environments, paintings and book covers). Nevertheless, these features showed positive results even on low textured scenarios (e.g. car exterior) in the automotive domain. This result brings insight about the extent of use of such features.
- The removal of undesired areas was introduced once it was realized that it can be decisive in order to achieve a successful reconstruction of the target model. By removing these areas the model generation algorithm deals with reduced ambiguity which can improve keypoint matching.
- It was also found that checking and giving feedback about the current tracking quality improves the user experience. The understanding of how well the system is handling each scenario and viewpoint gives the user the opportunity to better position the viewing angle in order to obtain a more precise augmentation. This communication between user and system induces one to help the other, and therefore collaborates for a better experience.

6. Conclusion

It was presented a tracking system based on natural features for AR applications targeted to the automotive domain. The system was evaluated during the Volkswagen/ISMAR Tracking Challenge 2014, and additional tests in a similar competition environment created by the authors were also performed.

In comparison with the most closely related existing expert and intelligent systems that focus on AR for the automotive domain, the proposed solution presents contributions, strengths and also some weaknesses. There are a number of existing solutions

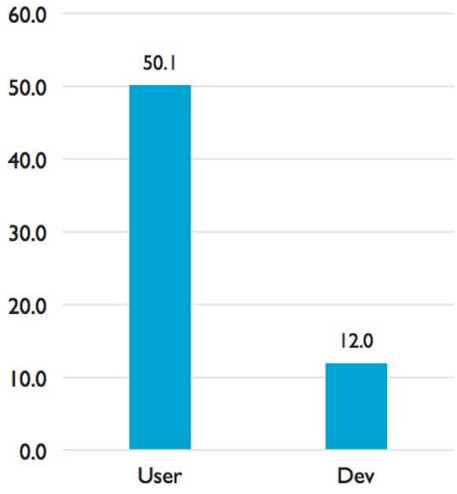

Fig. 23. Average tracking accuracy (in millimeters) by the two groups of competitors in recreated scenario 3.

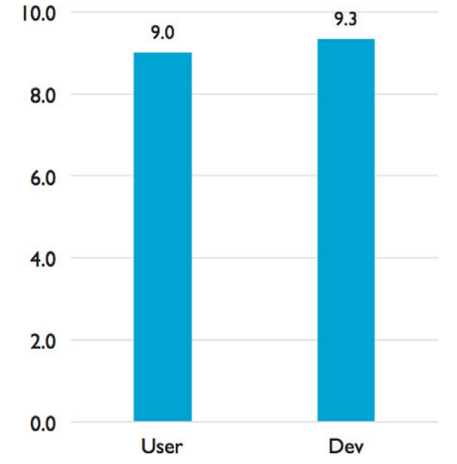

Fig. 24. Average time in minutes spent to identify challenge points by the two groups of competitors in recreated scenario 3.

that rely on markers to perform tracking (Lee & Rhee, 2008; Makris, Karagiannis, Koukas, & Matthaiakis, 2016; Makris et al., 2013; Nee et al., 2012; Reiners et al., 1998), while the proposed system is able to track the environment using its natural features. The proposed solution also covers several parts of an entire vehicle, while some existing systems focused only on one or a few specific parts, such as engine (Lee & Rhee, 2008; Nee et al., 2012; Platonov et al., 2006), differential (Makris et al., 2013), axle (Makris et al., 2016), girder (Nee et al., 2012), doorlock (Reiners et al., 1998) and exterior (Stanimirovic et al., 2014). Regarding the existing solutions based on natural features, the model generation step requires CAD data of the vehicle to be tracked and is performed manually (Stanimirovic et al., 2014) or with the aid of a marker (Platonov et al., 2006). In contrast, the proposed system allows automatic markerless model generation without the need of a CAD model. The method described in Stanimirovic et al. (2014) also performs manual tracking initialization, while the proposed solution is able to initialize tracking automatically. However, the techniques detailed in Platonov et al. (2006); Stanimirovic et al. (2014) better exploit temporal information by using a recursive tracking approach, which is not carried out by the proposed system. In addition, the tracking procedure adopted by Stanimirovic et al. (2014) takes into account both texture and edge information, whereas the proposed solution is based only on texture cues.

The system showed to be suitable for AR tasks in the automotive sector. The main positive aspect is that regular users are able to track the vehicle exterior and identify its parts. Fewer systems, such as Henderson and Feiner (2011); Porter et al. (2010), made tests with non-developer users in order to assert how well they performed using it. The combination of an automatic model generation and natural feature tracker is an important aspect that makes the proposed system easy to use by non-developer users. No other system combines these two characteristics for the automotive sector. Another strength is the high precision tracking. Fewer systems state their precision. For instance, Comport, Marchand, Pressigout, and Chaumette (2006) mention a smaller error. However, it is not designed for the automotive sector.

Current limitations of the proposed tracking system include: low frame rate when the number of 3D keypoints in the model is large; error accumulation when the entire vehicle is reconstructed in a single take; lack of temporal continuity, which may result in jittering; sensitivity to extreme illumination conditions; and occasional failures when dealing with scenes that have minimal texture information.

As future work, a method for selecting only the most relevant keyframes and keypoints will be investigated, in an attempt to reduce the model size and improve the frame rate. A large scale RGB-D based reconstruction approach such as the ones described in Chen, Bautembach, and Izadi (2013); Whelan et al. (2012) will be employed in order to have the entire vehicle in the same coordinate system without accumulating error. A recursive tracking method should also be used together with the detection technique presented in this work, in a similar way to what is done in Kim, Lepetit, and Woo (2010); Wagner, Schmalstieg, and Bischof (2009). This would allow taking benefit from both worlds: performance, accuracy and robustness of recursive tracking techniques and automatic initialization and recovery from failures of detection techniques. A hybrid approach that takes into account edge and depth information in addition to texture cues during tracking will also be investigated in order to better handle low textured and poorly illuminated scenes.

Acknowledgments

The authors would like to thank Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (processes 456800/2014-0, 140898/2014-0) for partially funding this research.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.eswa.2017.03.060

References

AG, V. (2014). Volkswagen tracking challenge 2014 - competition process. http://www.tracking-challenge.de/content/tc/content/en/history/challenge_2014/scenarios.html. [Online; accessed 31-January-2017].

Chen, J., Bautembach, D., & Izadi, S. (2013). Scalable real-time volumetric surface reconstruction. *ACM Transactions on Graphics (TOG), 32*(4), 113.

Comport, A., Marchand, E., Pressigout, M., & Chaumette, F. (2006). Real-time markerless tracking for augmented reality: the virtual visual servoing framework. *Visualization and Computer Graphics, IEEE Transactions on, 12*(4), 615–628. doi:10.1109/TVCG.2006.78.

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM, 24*(6), 381–395. doi:10.1145/358669.358692.

Georgel, P., Schroeder, P., Benhimane, S., Hinterstoisser, S., Appel, M., & Navab, N. (2007). An industrial augmented reality solution for discrepancy check. In *Proceedings of the 2007 6th IEEE and ACM international symposium on mixed and augmented reality*. In *ISMAR '07* (pp. 1–4). Washington, DC, USA: IEEE Computer Society. doi:10.1109/ISMAR.2007.4538834.

Henderson, S., & Feiner, S. (2011). Exploring the benefits of augmented reality documentation for maintenance and repair. *Visualization and Computer Graphics, IEEE Transactions on, 17*(10), 1355–1368. doi:10.1109/TVCG.2010.245.

Henderson, S. J., & Feiner, S. K. (2007). Augmented reality for maintenance and repair (armar). *Technical Report*. DTIC Document.

Horn, B. K. (1987). Closed-form solution of absolute orientation using unit quaternions. *JOSA A, 4*(4), 629–642.

Kato, H., & Billinghurst, M. (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Augmented reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM international workshop on* (pp. 85–94). IEEE.

Kim, K., Lepetit, V., & Woo, W. (2010). Scalable real-time planar targets tracking for digilog books. *The Visual Computer, 26*(6–8), 1145–1154.

Koller, D., Weber, J., & Malik, J. (1994). Robust multiple car tracking with occlusion reasoning. In J.-O. Eklundh (Ed.), *Computer vision ECCV '94*. In *Lecture Notes in Computer Science: 800* (pp. 189–196). Springer Berlin Heidelberg. doi:10.1007/3-540-57956-7_22.

Lee, J., & Rhee, G. (2008). Context-aware 3d visualization and collaboration services for ubiquitous cars using augmented reality. *The International Journal of Advanced Manufacturing Technology, 37*(5–6), 431–442. doi:10.1007/s00170-007-0996-x.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60*(2), 91–110.

Lv, Q., Josephson, W., Wang, Z., Charikar, M., & Li, K. (2007). Multi-probe lsh: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on very large data bases* (pp. 950–961). VLDB Endowment.

Makris, S., Karagiannis, P., Koukas, S., & Matthaiakis, A.-S. (2016). Augmented reality system for operator support in human–robot collaborative assembly. *CIRP Annals-Manufacturing Technology, 65*(1), 61–64.

Makris, S., Pintzos, G., Rentzos, L., & Chryssolouris, G. (2013). Assembly support using ar technology based on automatic sequence generation. *CIRP Annals-Manufacturing Technology, 62*(1), 9–12.

Marchand, E., Uchiyama, H., & Spindler, F. (2016). Pose estimation for augmented reality: a hands-on survey. *IEEE Transactions on Visualization and Computer Graphics, 22*(12), 2633–2651. doi:10.1109/TVCG.2015.2513408.

Moreno-Noguer, F., Lepetit, V., & Fua, P. (2007). Accurate non-iterative o(n) solution to the pnp problem. In *Computer vision, 2007. ICCV 2007. IEEE 11th international conference on* (pp. 1–8). doi:10.1109/ICCV.2007.4409116.

Nee, A., Ong, S., Chryssolouris, G., & Mourtzis, D. (2012). Augmented reality applications in design and manufacturing. *CIRP Annals-Manufacturing Technology, 61*(2), 657–679.

Newcombe, R. A., Davison, A. J., Izadi, S., Kohli, P., Hilliges, O., Shotton, J., Molyneaux, D., Hodges, S., Kim, D., & Fitzgibbon, A. (2011). Kinectfusion: Real–time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on* (pp. 127–136). IEEE.

Platonov, J., Heibel, H., Meier, P., & Grollmann, B. (2006). A mobile markerless ar system for maintenance and repair. In *Proceedings of the 5th IEEE and ACM international symposium on mixed and augmented reality* (pp. 105–108). IEEE Computer Society.

Porter, S., Marner, M., Smith, R., Zucco, J., & Thomas, B. (2010). Validating spatial augmented reality for interactive rapid prototyping. In *Mixed and augmented reality (ISMAR), 2010 9th IEEE international symposium on* (pp. 265–266). doi:10.1109/ISMAR.2010.5643599.

Reiners, D., Stricker, D., Klinker, G., & Mller, S. (1998). Augmented reality for construction tasks: Doorlock assembly. In *Proceedings of the IEEE and ACM IWAR98 (1. International workshop on augmented reality* (pp. 31–46). AK Peters.

Rentzos, L., Vourtsis, C., Mavrikios, D., & Chryssolouris, G. (2014). Using vr for complex product design. In *International conference on virtual, augmented and mixed reality* (pp. 455–464). Springer.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *Computer vision (ICCV), 2011 IEEE international conference on* (pp. 2564–2571). IEEE.

Salzmann, H., & Froehlich, B. (2008). The two-user seating buck: Enabling face-to-face discussions of novel car interface concepts. In *Virtual reality conference, 2008. VR '08. IEEE* (pp. 75–82). doi:10.1109/VR.2008.4480754.

Smeulders, A. W. M., Chu, D. M., Cucchiara, R., Calderara, S., Dehghan, A., & Shah, M. (2014). Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(7), 1442–1468. doi:10.1109/TPAMI.2013.230.

Stanimirovic, D., Damasky, N., Webel, S., Koriath, D., Spillner, A., & Kurz, D. (2014). [poster] a mobile augmented reality system to assist auto mechanics. In *Mixed and augmented reality (ISMAR), 2014 IEEE international symposium on* (pp. 305–306). doi:10.1109/ISMAR.2014.6948462.

Wagner, D., Schmalstieg, D., & Bischof, H. (2009). Multiple target detection and tracking with guaranteed framerates on mobile phones. In *Mixed and augmented reality, 2009. ISMAR 2009. 8th IEEE international symposium on* (pp. 57–64). IEEE.

Whelan, T., Kaess, M., Fallon, M., Johannsson, H., Leonard, J., & McDonald, J. (2012). Kintinuous: Spatially extended kinectfusion, Technical Report, 2012, MIT-CSAIL-TR-2012-020.

Wuest, H., Vial, F., & Strieker, D. (2005). Adaptive line tracking with multiple hypotheses for augmented reality. In *Mixed and augmented reality, 2005. proceedings. Fourth IEEE and ACM international symposium on* (pp. 62–69). doi:10.1109/ISMAR.2005.8.

Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. *ACM Computer Surveys, 38*(4). doi:10.1145/1177352.1177355.