HAL
open science

A Method to Quantitatively Evaluate Geo Augmented Reality Applications

Thibaud Michel, Pierre Genevès, Nabil Layaïda

To cite this version:

Thibaud Michel, Pierre Genevès, Nabil Layaïda. A Method to Quantitatively Evaluate Geo Augmented Reality Applications. ISMAR 2018 - International Symposium on Mixed and Augmented Reality, Oct 2018, Munich, Germany. pp.1-6. hal-01890838

HAL Id: hal-01890838

https://inria.hal.science/hal-01890838

Submitted on 9 Oct 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Method to Quantitatively Evaluate Geo Augmented Reality Applications

Thibaud Michel*, Pierre Genevès, and Nabil Layaïda
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, 38000 Grenoble, France

Abstract

We propose a method for quantitatively assessing the quality of Geo AR browsers. Our method aims at measuring the impact of attitude and position estimations on the rendering precision of virtual features. We report on lessons learned by applying our method on various AR use cases with real data. Our measurement technique allows to shedding light on the limits of what can be achieved in Geo AR with current technologies. This also helps in identifying interesting perspectives for the further development of high-quality Geo AR applications.

1 Introduction

Geo Augmented Reality (Geo AR) is an Augmented Reality (AR) method which allows the user to visualize augmented features exclusively thanks to the device position and orientation (Figure 1). In contrary to AR based on vision, this technique does not use image processing, is less computational expensive and does not rely on a specific lightning nor knowledge of the context. GPS, WiFi, Bluetooth or any kind of location sensors can be used to determine device position. The orientation of the device can be computed by an attitude estimation filter using the embedded inertial sensors (gyroscope, accelerometer and magnetometer).

Recently, several companies have started to develop applications which use Geo AR, including Layar [7], Peak.AR [12]. Based on user feedbacks [4], Geo AR applications could be much more immersive if they become more accurate.

Figure 1: Geo Augmented Reality.

A key activity in developing AR experiences is the evaluation of the quality and usability of the AR application. In their survey [3], Dünser et al. categorize AR papers containing a user evaluation according to the type of user study method and approaches.

They identified the following five main types of evaluation techniques: objective measurements, subjective measurements, qualitative analysis, usability evaluation techniques and informal evaluations. More generally, in [2, 4], authors evaluated the usefulness of an AR browser in an application.

To the best of our knowledge, all previous papers which evaluate Geo AR systems include the user in the loop. There does not exist any research work which quantitatively evaluates the precision of a Geo AR system. The main difficulty is that errors coming from both estimations (attitude and position) and are not of the same nature. This makes it difficult to combine them in order to provide an overall error estimation. We propose the first method for quantitatively assessing the quality of Geo AR browsers. This proposed method can be used in addition to existing methods [3] to evaluate the usability of a Geo AR application.

Contribution We propose a method for characterizing the impact of attitude and position estimations on the rendering precision of virtual features. This provides a quantitative measure of the quality of Geo AR applications. This makes it possible to identify criteria to better understand the limits of Geo AR for different use cases. We applied our method on 4 AR use cases with real data: an application to identify mountains and cities, an application to discover the history of a city, an application to make 3D models appear and turn around in an indoor environment, and finally an application to identify and interact with objects in a building using UWB (Ultra-WideBand). Our measurement technique allows to shedding light on the limits of what can be achieved in Geo AR with current technologies. This also helps in identifying interesting perspectives for the further development of accurate Geo AR applications.

2 Background

Geo AR is a technique which relies on two principal components: estimation of the device attitude and estimation of the position of the device. These estimations are computed from the data from inertial sensors and location sensors found in smartphones or Head Mounted Devices (HMD). In this section, we review the basic principles of attitude and position estimation.

2.1 Attitude estimation

The smartphone attitude is determined when the axis orientation of the Smartphone-Frame SF is specified with respect to the Earth-Frame EF (or Local Tangent Plane (LTP)). The SF_x-axis is horizontal and points to the right, the SF_y-axis is vertical and points up and the SF_z-axis points towards the outside of the front face of the screen. The EF_y-axis points to the North. The EF_z-axis points to the sky perpendicular to the reference ellipsoid and the EF_x-axis completes the right-handed coordinate system, pointing East (ENU : East, North, Up). Based on the literature, the attitude can be expressed with four different mathematical representations: Euler angles (yaw, pitch, roll), rotation matrices, quaternions or axis/angle.

The problem of finding the optimal attitude estimation solution was formulated for the first time by Wahba in 1965 [14]. Wahba's formulation seeks to find a rotation matrix between two coordinate systems from a set of vector observations (minimum two vectors known in a fixed frame and in a body frame). In our case, the two coordinate systems are the SF and the EF. A typical Inertial Measurement Unit fixed within a smartphone can provide two vector observations: acceleration in SF provided by an accelerometer noted ${}^{S}\mathrm{acc}$ and magnetic field in SF provided by a magnetometer noted ${}^{S}\mathrm{mag}$. If the smartphone is in static phase (not translating), and in a context without magnetic perturbation, acceleration and magnetic field vector observations are known in EF:

$$ {}^{E}\mathrm{acc} = \begin{bmatrix} 0 & 0 & g \end{bmatrix}^T, \quad {}^{E}\mathrm{mag} = \begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^T, \qquad (1) $$

*e-mail: thibaud.michel@inria.fr

where g is the gravity magnitude and m_x, m_y, m_z can be obtained using the WMM (World Magnetic Model) [13].

In addition to accelerometer and magnetometer, the gyroscope is usually used to estimate variation of attitude. Unfortunately, after integration, the gyroscope bias leads to an angular drift, increasing linearly over time ($\sim 5°/min$ when the sensor has been calibrated). Since the use of only gyroscope is not sufficient for attitude estimation, accelerometer and magnetometer are used to get an absolute quaternion and compensate for the drift. The crux in solving the attitude estimation problem finally consists in combining inertial and magnetic sensor measurements in a relevant manner to prevent magnetic perturbations and movements.

The precision of attitude estimation is crucial in Geo AR, as features should be seen at the right place on the screen.

2.2 Position estimation

Geolocation is the identification or estimation of the real-world geographic location of an object, it involves the generation of a set of geographic coordinates. Over the years, sensors like GPS, accelerometer, magnetometer, gyroscope, WiFi... have been miniaturized. In the beginning of 2010's, we observed a multitude of new sensors which can be embedded in recent devices (e.g. smartphones and HMD). These sensors make it possible to reuse geolocation techniques already developed in other fields like robotics and military.

We give below a brief overview of several geolocation techniques that can be used in a smartphone or HMD [9].

GNSS Usually, devices support U.S. GPS (Global Positioning System) as well as GLONASS (Russian Global Navigation Satellite System). U.S. government claims an horizontal accuracy of less than 1.9 meters 95% of the time [11] when it is used in a clear space. Nevertheless, GNSS relies on time of flight values, thus, when the line of sight between receiver and a satellite is obstructed by an object, for instance a building, accuracy is degraded.

WiFi Fingerprinting For WiFi fingerprinting it is assumed that each location in a building can be identified by its unique Fingerprint. A WiFi Fingerprint is the (in theory) unique combination of the Received Signal Strength Indicator (RSSI) and its according Access point (AP, represented by its unique MAC address). This system does not require line of sight, the knowledge about the exact APs locations and therefore neither angle nor distance measurements. WiFi Fingerprinting is usually divided into two phases: An offline (site survey) phase and an online (navigation) phase.

WiFi Trilateration A second technique to determine user position with WiFi signals consists in using trilateration with RSSI from WiFi AP. Trilateration is a widely used technique to determine position with radio signals: GPS, Bluetooth, UWB, etc. Trilateration is the process of determining locations of points by measurement of distances, using the geometry circles.

SHS The Step and Heading System (SHS) is a Pedestrian Dead Reckoning (PDR) technique which consists in detecting user's step, estimating step size and user direction. In a simple implementation, the user holds the phone in front of him and each step causes position to move forward a "step length" distance in the direction measured by the compass. The main drawback of a PDR based technique is that it requires the knowledge of the starting position.

Ultra-Wideband (UWB) One more recently developed wireless technology is Ultra-Wideband (UWB) radio. UWB is used to communicate between devices, similar to Bluetooth and WiFi, but with a higher data rate. UWB has also been designed specifically to transmit in a way that enables much more precise distance measurements and location positioning. As of 2018, commodity smartphones are not yet equipped with a UWB receiver.

Precision of device positioning is vital, because the pair <position, orientation> defines where features will be displayed on the screen.

3 A Quantitative Method to Evaluate Geo AR

In this section, we present a new evaluation method whose aim is to characterize the quality of Geo AR browsers through a set of quantitative measurements.

As we have seen earlier, the Geo AR approach crucially relies on the estimations of attitude and position. The (im)precision of these estimations may greatly affect the user experience of the Geo AR application in various manners. It is thus not easy to capture the impact of a poor attitude estimation or a poor positioning estimation on the rendering performed by an AR browser. One reason for this, is that errors from positioning estimation and errors from attitude estimation are not of the same nature. This makes it hard to combine them directly. We present the first method that seeks to quantitatively exploit these errors for providing an overall meaningful measure that characterizes the final error observed in the AR scene on the screen.

We introduce the notion of distance between the virtual feature and the device. This parameter is very important, because the impact on the rendering can be totally different depending on whether a feature is close or far.

3.1 General Principle

We propose an evaluation model to calculate the average distance between a real and a virtual point represented on the screen given the 2 vectors of errors (F_{pos} and F_{att}):

- f_{pos} denotes a positioning estimation error in meters given by a navigation algorithm (see Section 2.2). F_{pos} is the vector of f_{pos} values for a specific use case (e.g: outdoor using GNSS).
- f_{att} denotes an attitude estimation error obtained from a attitude estimation filter (see Section 2.1). F_{att} is the vector of f_{att} values for a specific use case (e.g: precision error of an attitude estimation filter during high magnetic perturbations).

We then consider a feature point (P) at a fixed distance (d) (see Figure 2). The feature (here: a tree) is displayed in the middle of the smartphone (S) screen. The system estimates the position of the feature (P') from attitude and position of the device. v is the distance between the estimated feature (P') and the real position of the feature (P). The projection of this distance on the screen is named e. This distance depends on screen size l and the field of view (fov) of the rendering (for the sake of clarity fov is not rendered on the figure). And finally, ϕ is the positive angle between $\vec{SP}$ and $\vec{SP'}$. The process explained here is the same as an OpenGL rendering for a 3D scene [9]. In [5,6], to evaluate visual SLAM techniques for AR, some metrics are equivalent to ours. v is equivalent to 3DEVO (3D Error of Virtual Object), and e is equivalent to PEVO (Projection Error of Virtual Object).

Figure 2: Representation of errors due to a poor estimation of a virtual feature position. v is the distance between the estimated feature (P') and the real position of the feature (P). The projection of this distance on the screen is named e.

We start by reducing the problem to look for parameters v and ϕ because they are independent to the screen size and the field of view.

3.2 Evaluation: a Varying Attitude with a Fixed Position

As a first step, we assume the estimated position perfect ($f_{\text{pos}} = 0$). The experimental study found in [10] identifies a specific behavior of attitude filters during AR motions. We notice that errors on pitch and roll angles are mainly due to the presence of external accelerations during the estimation phase. If we consider a Point of Interest (PoI) in an AR application, errors on pitch and roll angles correspond to a PoI which will not be placed at the correct elevation (pitch-angle) and which will suffer from a rotation (roll-angle). Since typical AR motions are rather slow, external accelerations are rather low and attitude estimation tends to yield more accurate results. During magnetic perturbations, the approach from Martin et al. [8] helps to avoid the impact of the measurement on the pitch and roll. In accordance with [10], we can thus consider pitch and roll angles well estimated during all kinds of AR motions. However, the yaw angle remains impacted during magnetic perturbations, the consequence will be a PoI misplaced on the horizon line ($P'_z = 0$).

We consider f_{att}, the angle between $\vec{SP}$ and $\vec{SP'}$ due to a poor attitude estimation. To determine $v_{f_{\text{pos}}=0}$ and $\phi_{f_{\text{pos}}=0}$, we propose the following model:

$$\begin{cases} S = (0,0,0) \\ P = (0,d,0) \end{cases} \qquad \begin{cases} \widehat{PSP'} = f_{\text{att}},\ P'_z = 0 \\ \|SP'\| = d \end{cases} \tag{2}$$

We consider α, the angle between $P'_{f_{\text{pos}}=0}$, $C_{f_{\text{pos}}=0}$ and the x-y plan where $C_{f_{\text{pos}}=0}$ is the projection of P' on $[SP]$. Since attitude estimation error boils down to yaw-angle error [10], $\alpha \in \{0, \pi\}$.

(a) α fixed **(b)** α varying

Figure 3: Evaluation model and geometric illustration where $f_{\text{pos}} = 0$ and f_{att} is fixed.

With the modeling introduced above we determine the position of $C_{f_{\text{pos}}=0}$ and $P'_{f_{\text{pos}}=0}$:

$$\begin{aligned} C_{f_{\text{pos}}=0}(d, f_{\text{att}}) &= (0, d * cos(f_{\text{att}}), 0) \\ P'_{f_{\text{pos}}=0}(d, f_{\text{att}}, \alpha) &= C_{f_{\text{pos}}=0} + (d * sin(f_{\text{att}}) * cos(\alpha),\ 0,\ 0) \end{aligned} \tag{3}$$

Then we determine the distance $v_{f_{\text{pos}}=0}$ and the angle $\phi_{f_{\text{pos}}=0}$:

$$\begin{aligned} \phi_{f_{\text{pos}}=0}(d, f_{\text{att}}) &= f_{\text{att}} \\ v_{f_{\text{pos}}=0}(d, f_{\text{att}}, \alpha) &= \sqrt{(d*sin(f_{\text{att}})*cos(\alpha))^2 + (d*cos(f_{\text{att}}) - d)^2\ )^2} \\ &= \sqrt{2 * d^2 (1 - cos(f_{\text{att}}))} \end{aligned} \tag{4}$$

The representation of the model and a geometric illustration are given in Figure 3.

3.3 Evaluation: a Varying Position with a Fixed Attitude

As a second step, we now assume attitude to be perfectly estimated ($f_{\text{att}} = 0$). We consider f_{pos}, the distance between the estimated position of the device ($S'_{f_{\text{att}}=0}$) and its real position (S).

To determine $v_{f_{\text{att}}=0}$ and $\phi_{f_{\text{att}}=0}$, we propose the following model:

$$\begin{cases} S = (0,0,0) \\ P = (0,d,0) \end{cases} \qquad \begin{cases} \|SS'\| = f_{\text{pos}} \\ \vec{S'P'} = \vec{SP} \end{cases} \tag{5}$$

$S'_{f_{\text{att}}=0}$ belongs to the sphere centered in S with a radius of f_{pos}. We can characterize S' with two parameters: γ and β. Let $S''_{f_{\text{att}}=0}$ be the projection of $S'_{f_{\text{att}}=0}$ in the x-y plan, γ is the angle between SS'' and the x-axis and β is the angle between $S''_{f_{\text{att}}=0}$, S and $S'_{f_{\text{att}}=0}$.

(a) β and γ fixed **(b)** β and γ varying

Figure 4: Evaluation model and geometric illustration where $f_{\text{att}} = 0$ and f_{pos} is fixed.

With this modeling we have:

$$\begin{aligned} S'_{f_{\text{att}}=0}(d, f_{\text{pos}}, \beta, \gamma) &= (f_{\text{pos}} * cos(\beta) * cos(\gamma), \\ &\qquad f_{\text{pos}} * cos(\beta) * sin(\gamma), f_{\text{pos}} * sin(\beta)) \\ P'_{f_{\text{att}}=0}(d, f_{\text{pos}}, \beta, \gamma) &= S' + (0, d, 0) \end{aligned} \tag{6}$$

Then we determine the distance $v_{f_{\text{att}}=0}$ and the angle $\phi_{f_{\text{att}}=0}$:

$$\begin{aligned} (\vec{P} - \vec{S}) \cdot (\vec{P'} - \vec{S}) &= \left\|\vec{P} - \vec{S}\right\| * \left\|\vec{P'} - \vec{S}\right\| * cos(\phi) \\ \vec{P} \cdot \vec{P'} &= d * \left\|\vec{P'}\right\| * cos(\phi) \end{aligned} \tag{7}$$

$$\begin{aligned} \phi_{f_{\text{att}}=0}(d, f_{\text{pos}}, \beta, \gamma) &= acos\left(\frac{\vec{P} \cdot \vec{P'}}{d * \left\|\vec{P'}\right\|}\right) = acos\left(\frac{P'_y}{\left\|\vec{P'}\right\|}\right) \\ v_{f_{\text{att}}=0}(d, f_{\text{pos}}, \beta, \gamma) &= f_{\text{pos}} \end{aligned} \tag{8}$$

The representation of the model and a geometric illustration are given in Figure 4.
Here, we considered f_{pos} as an error on the 3 dimensions. But, the model can be modified to handle distributions on 2D or 2.5D if vertical (f^v_{pos}) and horizontal (f^h_{pos}) errors are known:

$$S'_{2D}(f^h_{\text{pos}}) = (f^h_{\text{pos}} * cos(\gamma), f^h_{\text{pos}} * sin(\gamma), 0) \tag{9}$$

$$\begin{aligned} S'_{2.5D}(f^h_{\text{pos}}, f^v_{\text{pos}}) &= (f^h_{\text{pos}} * cos(\gamma), f^h_{\text{pos}} * sin(\gamma), f^v_{\text{pos}} * z) \\ &\text{where: } z \in \{-1, 1\} \end{aligned} \tag{10}$$

3.4 Evaluation: Varying Position and Attitude

We now consider that both attitude and position are not perfectly estimated (as it is the case in practice). We explain how the models can be combined. To determine v and ϕ, we propose the overall following model:

$$\begin{cases} S = (0,0,0) \\ P = (0,d,0) \\ \|SS'\| = f_{\text{pos}} \end{cases} \qquad \begin{cases} \widehat{PSP'} = f_{\text{att}} \\ \left\|\vec{S'P'}\right\| = d \end{cases} \tag{11}$$

(a) α, β and γ fixed **(b)** α, β and γ varying

Figure 5: Evaluation model where f_{att} and f_{pos} are fixed.

First, we determine positions of S', C and P':

$$\begin{aligned} S'(d, f_{\text{pos}}, \beta, \gamma) &= (f_{\text{pos}} * cos(\beta) * cos(\gamma), \\ &\quad f_{\text{pos}} * cos(\beta) * sin(\gamma), f_{\text{pos}} * sin(\beta)) \\ C(d, f_{\text{pos}}, f_{\text{att}}, \beta, \gamma) &= S' + (0,\ d * cos(f_{\text{att}}),\ 0) \\ P'(d, f_{\text{pos}}, f_{\text{att}}, \alpha, \beta, \gamma) &= C + (d * sin(f_{\text{att}}) * cos(\alpha),\ 0,\ 0) \end{aligned} \tag{12}$$

where α, β and γ are determined in the previous models. Then we determine the distance v and the angle ϕ:

$$\begin{aligned} \phi(d, f_{\text{pos}}, f_{\text{att}}, \alpha, \beta, \gamma) &= acos(\frac{P'_y}{\left\|\vec{P'}\right\|}) \\ v(d, f_{\text{pos}}, f_{\text{att}}, \alpha, \beta, \gamma) &= \left\|P'P\right\| \end{aligned} \tag{13}$$

The representation of the model and a geometric illustration are given in Figure 5.

3.5 Projected Distance on the Screen

In the previous section, we proposed a model to represent distance error (v) and angle error (ϕ) between the estimated feature and the real feature. So far this does not allow us to know how far will the virtual feature be displayed compared to the real one on the screen of the device. This distance (e) depends on two additional parameters: screen size (l) and the field of view (fov). We consider $H(H_x, H_z)$, the projection of $P'(P'_x, P'_y, P'_z)$ on the screen of the device (see Figure 6) and consequently: $e = \sqrt{H_x^2 + H_z^2}$. For the sake of clarity, in this section we will write P' as a shorthand for $P'(d, f_{\text{pos}}, f_{\text{att}}, \alpha, \beta, \gamma)$.

(a) Projection on z-axis **(b)** Projection on x-axis

Figure 6: Projection of distance error on the screen

In the following equations, we consider l_w and l_h, respectively the width and the height of the camera view on the device screen. The relation between l_w and l_h is expressed with the aspect ratio ar:

$$l_h = ar * l_w. \tag{14}$$

Here, we also consider that the aspect ratio of the camera sensor is also ar, to avoid distortions. fov_H and fov_V are respectively the horizontal and the vertical fields of view of the camera. The relation between fov_H and fov_V is given by:

$$\text{fov}_V = 2 * atan(tan(\frac{\text{fov}_H}{2}) * ar) \tag{15}$$

Now, we are looking for H given P', l and fov:

$$\begin{aligned} H_y(\text{fov}_H, l_w) &= \frac{l_w}{2 * tan(\frac{\text{fov}_H}{2})} \quad \text{and} \quad \theta_1 = atan(\frac{P'_x}{P'_y}) \\ H_x(P', \text{fov}_H, l_w) &= tan(\theta_1) * H_y(\text{fov}_H, l_w) \\ &= \frac{P'_x}{P'_y} * \frac{l_w}{2 * tan(\frac{\text{fov}_H}{2})} \end{aligned} \tag{16}$$

$$\begin{aligned} H_y(\text{fov}_V, l_h) &= \frac{l_h}{2 * tan(\frac{\text{fov}_V}{2})} \quad \text{and} \quad \theta_2 = atan(\frac{P'_z}{P'_y}) \\ H_z(P', \text{fov}_V, l_h) &= tan(\theta_2) * H_y(\text{fov}_V, l_h) \\ &= \frac{P'_z}{P'_y} * \frac{l_h}{2 * tan(\frac{\text{fov}_V}{2})} \end{aligned} \tag{17}$$

We will show that H_x can be expressed in terms of either the parameters $< \text{fov}_H, l_w >$ or with the parameters $< \text{fov}_V, l_h >$. We use Eq. (14) and (15) to replace l_w and fov_H in the right-hand side of Eq. 16 which becomes:

$$\begin{aligned} &= \frac{P'_x}{P'_y} * \frac{l_h}{ar} * \frac{1}{2 * tan(\frac{1}{2} * 2 * atan(tan(\frac{\text{fov}_V}{2}) * \frac{1}{ar}))} \\ &= \frac{P'_x}{P'_y} * \frac{l_h}{ar} * \frac{1}{2 * tan(\frac{\text{fov}_V}{2}) * \frac{1}{ar}} \\ &= \frac{P'_x}{P'_y} * \frac{l_h}{2 * tan(\frac{\text{fov}_V}{2})} \quad = \quad H_x(P', \text{fov}_V, l_h) \end{aligned} \tag{18}$$

This shows that $H_x(P', \text{fov}_H, l_w) = H_x(P', \text{fov}_V, l_h)$. Consequently, H_x and H_z can be expressed with the same pair of <field of view, screen size> (horizontal or vertical). Finally, the distance e which corresponds to the projected distance between the virtual and the real feature on the screen is given by:

$$e(P', fov, l) = \sqrt{H_x^2 + H_z^2} = \frac{\sqrt{P_x'^2 + P_z'^2}}{P'_y} * \frac{l}{2 * tan(\frac{fov}{2})} \tag{19}$$

4 Experimental Results

The distance v between the estimated feature and the real feature, as well as its projection e will allow us to quantify the usability of a Geo AR application. For example, is a Geo AR system accurate enough to turn around a 3D virtual object in a mall? If not, what precision is needed? To answer these kind of questions, as well as to select the most suited techniques for AR, we instantiate our model with results from attitude and positioning estimation obtained from benchmarks in [10] and [9].

4.1 Best Filters for Attitude Estimation in AR

In [10], 10 representative attitude filters were analyzed, and their precision were compared with a fine-grained setup in a motion-lab.

Table 1 summarizes the precision results obtained when a smartphone is used for AR (slow movements). The table is split in two columns: (i) estimation with few magnetic perturbations, which correspond to perturbations found in urban canyons and (ii) estimation with high magnetic perturbations typically found inside a building.

Several attitude filters have been compared. *Built-in* corresponds to the average precision obtained with attitude filters embedded in

	Urban canyons	Indoor
Built-in	7.1°	29.0°
Best-of-literature	4.5°	14.6°
Proposed filter	4.5°	9.8°

Table 1: Precision of attitude estimation according to Augmented Reality motions with few and high magnetic perturbations.

iOS and Android smartphones. It exhibits a precision error of 7.1° with few magnetic perturbations and 29° with high perturbations. Computing measures from inertial sensors using the best filter from the literature (*Best-of-literature*) reduces the average precision error from 7.1° to 4.5°. Finally, in [10], a new filter is designed to limit the impact of magnetic perturbations. This filter enhances the precision of 180% compared to the built-in filter and by 35% compared to the best of the literature. The motion capture has not been set up outside the building in a clear space, but, the expected precision is obviously lower than the 4.5° obtained with the low magnetic perturbations and should reach 1° or 2°.

4.2 Best Algorithms for Positioning Estimation in AR

In [9], several geolocation techniques have been compared in a common context. Trials have been conducted in 2 places: a 15 000 m^2-building and an outside 5 000 m^2-clear space area.

	Indoor		Outdoor	
	AVG	STD	AVG	STD
SHS	8.2 m	5.0 m	16.7 m	14.4 m
SHS + Map-Matching	2.3 m	1.6 m	11.9 m	9.6 m
WiFi-Fingerprinting	8.1 m	8.6 m	x*	x
WiFi-Trilateration	7.7 m	8.3 m	x*	x
UWB	0.5 m	0.3 m	x*	x
GNSS	25.4 m	14.8 m	3.5 m	2.6 m

* Technologies based on WiFi and UWB have not been deployed outside.

Table 2: Average (AVG) and Standard Deviation (STD) of precision error of navigation algorithms inside and outside.

Table 2 presents the results of six geolocation techniques. The results are obtained when the smartphone is held in the same orientation than the navigation frame. Indoor, UWB clearly outperforms other technologies. Unfortunately, UWB is the only technique we compared which is not natively implemented in smartphones. Among others, SHS + Map-Matching exhibits a good behavior with an average of 2.28 m, the gain compared to SHS without Map-Matching is 240%. Outdoor, GNSS has a mean error of 3.29 m and outperforms SHS techniques.

4.3 Adaptation of the Method for Data from Benchmarks

We have slightly adapted the evaluation method proposed previously to take into account data learned from the benchmarks:

- In Section 4.2, f_{pos} was calculated on 2 dimensions. The results obtained in [9] does not allow us to know positioning error vertically. For this reason, we used the 2D model proposed in Eq 9, therefore, $e_{2\text{D}}(d, f_{\text{pos}}, f_{\text{att}}, \alpha, \gamma, fov, l) = e(d, f_{\text{pos}}, f_{\text{att}}, \alpha, 0, \gamma, fov, l)$.
- In Section 4.1, the estimated attitude error was given using Quaternion Angle Difference (QAD). Theoretically, f_{att} is not equal to the attitude error, because QAD does not only represent angle between both vectors ($\vec{SP}$ and $\vec{SP'}$) but also the rotation of the feature around $\vec{SP}$-axis. As external accelerations are very low during AR motions, the pitch and roll rotations will be little affected, so we will consider $f_{\text{att}} \simeq$ yaw error.

We then propose to see how accurate the Geo AR applications are on different use cases. Results for each use case will be detailed with the average (AVG) and standard deviation (STD) of distances e and v (see Fig 2). As e, is dependent to the size of a device and the camera FOV, we fixed them to $l = 11.4\ cm$ and $fov = 60°$ which correspond to the metrics of a Nexus 5X. The problem is that in our benchmark we do not know in which direction the bad estimated attitude is pointing (α), we only know its magnitude (f_{att}). We also do not know in which direction the bad estimated position is (γ), we only know its magnitude (f_{pos}). From this information, we consider $E_{2\text{D}}$ be the average value of $e_{2\text{D}}$:

$$E_{2\text{D}}(d, f_{\text{pos}}, f_{\text{att}}, fov, l) = \frac{\sum_{\alpha \in \{0,\pi\}} \left(\int_{-\pi}^{\pi} e_{2\text{D}}(d, f_{\text{pos}}, f_{\text{att}}, \alpha, \gamma, fov, l)\ d\gamma \right)}{2 \int_{-\pi}^{\pi} 1\ d\gamma} \quad (20)$$

For its implementation, we used `integral` function of MATLAB. Average (μ) and standard deviation (σ) of E from a vector of position errors (F_{pos}) and a vector of attitude errors (F_{att}) are defined by:

$$\mu_E(d, F_{\text{pos}}, F_{\text{att}}, fov, l) = \frac{\sum_{f_{\text{pos}} \in F_{\text{pos}}} \left(\sum_{f_{\text{att}} \in F_{\text{att}}} \left(E_{2\text{D}}(d, f_{\text{pos}}, f_{\text{att}}, fov, l) \right) \right)}{\sum_{F_{\text{pos}}} \sum_{F_{\text{att}}}} \quad (21)$$

$$\sigma_E(d, F_{\text{pos}}, F_{\text{att}}, fov, l) = \sqrt{\frac{\sum_{f_{\text{pos}} \in F_{\text{pos}}} \left(\sum_{f_{\text{att}} \in F_{\text{att}}} \left(\left(E_{2\text{D}}(d, f_{\text{pos}}, f_{\text{att}}, fov, l) - \mu_e \right)^2 \right) \right)}{\sum_{F_{\text{pos}}} \sum_{F_{\text{att}}}}} \quad (22)$$

In the same manner, we defined $\mu_{V_{2\text{D}}}$, $\sigma_{V_{2\text{D}}}$, the average and standard deviation of v from a vector of position errors (F_{pos}) and a vector of attitude errors (F_{att}) In the rest of the paper, for the sake of clarity, we consider $e = E_{2\text{D}}$, $v = V_{2\text{D}}$. We then evaluated 4 typical use cases of Geo AR.

4.4 Application to Four Use Cases

We chose to apply our evaluation method on four use cases where an AR application can be useful for entertainment or tourism. Two of them take place indoor and the two others outdoor. We also varied the distance between the user position and the virtual feature position according to real situtations.

4.4.1 Use Case 1: Mountains identification

From a high position, a person wants to identify mountains around him. The person is on a hiking trail or in a ski resort and the space around him is clear. Considered **position** from GNSS outside is $\simeq 3.54\ m$. **Attitude** from AR with a low magnetic pert. is $\simeq 4.5°$. In reality, it should be less, because our benchmark was setup in a building and we noticed some magnetic perturbations. Finally, considered **distances to feature** are from 1 km to 50 km. Table 3

	Screen distance (e)		Real to virtual distance (v)	
	AVG	STD	AVG	STD
Feature at 1 km	0.8 cm	0.0 cm	78.6 m	2.5 m
Feature at 10 km	0.8 cm	0.0 cm	785.2 m	2.5 m
Feature at 50 km	0.8 cm	0.0 cm	3926.0 m	2.5 m

Table 3: Scores of: mountains identification.

exhibits results for 3 distances of features (1 km, 10 km and 50 km). The average distance error on the screen is limited, and even smaller for attitude error of 2° (e reduces to 0.3 cm). Conversely, if the positioning error grows to 50 m; e and v remain almost unchanged. Notice that v grows linearly in function of the feature distance. This means that, if the user aims at a 50 km-far summit, there is a significant risk that another mountain name (e.g. 4 km away from

the one which is aimed) is displayed. Notice that in that case, more comprehensive usability studies [3] are useful in addition to our method. Finally, if we consider the user in a building instead of a clear space ($f_{att} = 9.8°$ and $f_{pos} = 25\ m$) and place a feature at $10\ km$: $e = 1.9\ cm$ and $v = 1882\ m$. This shows the limits of aiming a far-feature from a place impacted by magnetic perturbations.

4.4.2 Use Case 2: City History Tour

A person is touring a city. He wants to learn more about the history of this city. He uses his smartphone to read stories by pointing it to old buildings. Considered **position** from GNSS in downtown is $\simeq 15\ m$ [1]. **Attitude** from AR with a low magnetic pert. is $\simeq 4.5°$. Finally, considered **distances to feature** are from $5\ m$ to $100\ m$. Table 4 shows results for 4 distances of features ($5\ m$, $20\ m$, $30\ m$ and $100\ m$). In this use case, it is not possible to provide a good AR experience if the feature is too close from the user. But, if building is big enough ($> 30\ m$-wide), from $d = 30\ m$, AR starts to be more reliable because, the feature information will be shown as an overlay of the building. This is true only if the building size is greater than $2 * v$. We also tried the $30 - m$ far use case with a better estimation of attitude $f_{att} = 1°$ and results obtained are similar. However, if we decrease f_{pos} to 5 meters, when $d = 30\ m$, the projected error e on the screen is equal to $1.2\ cm$. This shows that to enhance reliability of this use case, more efforts needs to be made on the positioning estimation than the attitude estimation.

	Screen distance (e)		Real to virtual distance (v)	
	AVG	STD	AVG	STD
Feature at $5\ m$	$+\infty^*$	$+\infty^*$	$15.0\ m$	$0.3\ m$
Feature at $20\ m$	$6.2\ cm$	$3.8\ cm$	$15.1\ m$	$1.1\ m$
Feature at $30\ m$	$3.5\ cm$	$1.9\ cm$	$15.1\ m$	$1.7\ m$
Feature at $100\ m$	$1.1\ cm$	$0.8\ cm$	$16.0\ m$	$5.4\ m$

* e is not provided when $f_{pos} > d$ because P' is not projected on the screen.

Table 4: Scores of: city history tour.

4.4.3 Use Case 3: Turn Around 3D Models Indoor

In a building, a user makes a 3D model appear (e.g.: a cat). Then, he turns around to look the 3D model from other angles. Considered **position** from SHS + Map-Matching is $\simeq 2.3\ m$. **Attitude** from AR with a high magnetic pert. is $\simeq 9.8°$. Finally, considered **distances to feature** are from $0.5\ m$ to $2\ m$. Table 5 shows the results for 3 distances of features ($0.5\ m$, $1\ m$ and $2\ m$). It is clearly shown that using the Geo AR for this kind of application is not relevant. The virtual feature will almost never be displayed at the right place on the screen because $f_{pos} > d$. As long as f_{pos} is not significantly higher than the distance to the feature (d), this AR use case remains not reliable.

	Screen distance (e)		Real to virtual distance (v)	
	AVG	STD	AVG	STD
Feature at $0.5\ m$	$+\infty^*$	$+\infty^*$	$2.3\ m$	$0.1\ m$
Feature at $1\ m$	$+\infty^*$	$+\infty^*$	$2.3\ m$	$0.1\ m$
Feature at $2\ m$	$+\infty^*$	$+\infty^*$	$2.3\ m$	$0.3\ m$

* e is not provided when $f_{pos} > d$ because P' is not projected on the screen.

Table 5: Scores of: make 3D models appear and turn around in an indoor environment.

4.4.4 Use Case 4: Interact with Objects in a Building

A user points objects in a house to monitor the energy consumption (e.g. radiators) or to interact with them (e.g. lights, blinds). Considered **position** from UWB is $\simeq 0.5\ m$. **Attitude** from AR with a high magnetic pert. is $\simeq 9.8°$. Finally, considered **distances to feature** are from $0.5\ m$ to $5\ m$. Table 6 exhibits results for 4 distances of features ($0.5\ m$, $1\ m$, $2\ m$ and $5\ m$). The results are averaged. This use case mostly works when features are far. We also tried the $1 - m$ far use case with a better estimated attitude $f_{att} = 2°$ (divided by 5) and we obtained $e = 3.4\ cm$. The difference with the previous e is not very noticeable. However, if we decrease f_{pos} to 0.25 meters (divided by 2), we obtain $e = 2.1\ cm$. Once again, to improve the accuracy of AR for this use case, efforts must be made on the positioning estimation.

	Screen distance (e)		Real to virtual distance (v)	
	AVG	STD	AVG	STD
Feature at $0.5\ m$	$36.0\ cm$	$105.9\ cm$	$0.5\ m$	$0.1\ m$
Feature at $1\ m$	$3.7\ cm$	$2.5\ cm$	$0.5\ m$	$0.1\ m$
Feature at $2\ m$	$2.1\ cm$	$1.6\ cm$	$0.6\ m$	$0.2\ m$
Feature at $5\ m$	$1.9\ cm$	$0.7\ cm$	$1.0\ m$	$0.3\ m$

Table 6: Scores of: identify and interact with objects in a building using UWB.

5 Conclusion

The usability of a Geo AR application is related to many factors [3]. We propose an evaluation method which is based on a quantitative and objective measure of the perceived error in AR rendering. We use our method to evaluate the suitability of different techniques for the construction of Geo AR applications. This gives insights on the feasibility and the usability of Geo AR applications built with current technologies. For example, we show that, in some specific contexts, building Geo AR applications is still beyond reach with current techniques (e.g. turning around a 3D model would require a significant progress on the positioning estimation). The present work also helps in identifying interesting perspectives for the further development of high-quality Geo AR applications. For example, some Geo AR applications such as monitoring objects indoor with UWB are particularly promising.

References

[1] B. Ben-Moshe, E. Elkin, H. Levi, and A. Weissman. Improving accuracy of gnss devices in urban canyons. In *CCCG*, 2011.

[2] A. Dey, M. Billinghurst, R. W. Lindeman, and J. Swan. A systematic review of 10 years of augmented reality usability studies: 2005 to 2014. *Frontiers in Robotics and AI*, 5:37, 2018.

[3] A. Dünser, R. Grasset, and M. Billinghurst. *A survey of evaluation techniques used in augmented reality studies*. HIT-Lab, NZ, 2008.

[4] A. Dünser, M. Billinghurst, J. Wen, V. Lehtinen, and A. Nurminen. Exploring the use of handheld AR for outdoor navigation. *Computers & Graphics*, 36(8):1084–1095, Dec. 2012.

[5] R. Ichikari, T. Kurata, K. Makita, T. Taketomi, H. Uchiyama, T. Kondo, S. Mori, and F. Shibata. Reference Framework on vSRT-method Benchmarking for MAR. In *ICAT-EGVE*. The Eurographics Assoc., 2017.

[6] ISO 18520. Benchmarking of vison-based spatial registration and tracking methods for Mixed and Augmented Reality (MAR). Technical report, International Organization for Standardization, Nov. 2017.

[7] Layar Team. Layar. `https://www.layar.com`, 2009–2018.

[8] P. Martin and E. Salaün. Design and implementation of a low-cost observer-based attitude and heading reference system. *Control Engineering Practice*, 18(7):712–722, 2010.

[9] T. Michel. *On Mobile Augmented Reality Applications based on Geolocation*. PhD thesis, Université Grenoble Alpes, 2017.

[10] T. Michel, P. Genevès, H. Fourati, and N. Layaïda. Attitude estimation for indoor navigation and augmented reality with smartphones. In *Pervasive and Mobile Computing Journal (PMCJ)*. Elsevier, 2018.

[11] National Coordination Office for Space-Based Positioning, Navigation, and Timing. How accurate is gps? `http://www.gps.gov/systems/gps/performance/accuracy/`, 2017.

[12] F. Soldati. Peak AR. `http://www.peakfinder.org/mobile`, 2018.

[13] U.S. National Geospatial-Intelligence Agency (NGA) and the U.K.'s Defence Geographic Centre (DGC). The world magnetic model, 2015.

[14] G. Wahba. A least squares estimate of satellite attitude. *SIAM review*, 7(3):409–409, 1965.