High Precision Object Geo-Localization and Visualization in Sensor Networks

Simon Lemaire[a], Christoph Bodensteiner[a], Michael Arens[a]

[a]Fraunhofer IOSB, Gutleuthausstr. 1, Ettlingen, Germany

ABSTRACT

The wide availability of previously acquired, geo-referenced imagery enables automatic video based solutions for high precision object geo-localization and cooperative visualization. We present a system which geo-references objects seen in UAV video streams, distributes this information in a sensor network and visualizes them on modern smartphones using augmented reality techniques.

The feasibility of the approach was experimentally validated using Mini-UAV ("MD-400") and high altitude UAV video footage in combination with modern off-the-shelve smartphones.

Applications are widespread and include for instance crisis and disaster management or military applications.

Keywords: Object Localisation, Geo-Referentiation, Image Registration, Visual Self-Localization

1. INTRODUCTION

A fundamental task in remote sensing and computer vision is the precise geo-localization and the image transfer of objects seen in camera images. A elegant solution to this problem is the determination of a geometric transformation (e.g. image registration) between the camera image and image material which has already been geo-referenced. The wide availability of such previously acquired, geo-referenced imagery (e.g. Google Maps) enables such automatic solutions for high precision object geo-localization, sensor hand-over and visualization.

First we precisely geo-localize objects seen in UAV video stream. The geo-localized object is then transmitted to mobile devices on the ground and subsequently visualized using augmented reality techniques. This approach enables the transfer of object information with low bandwidth connections since we only transfer highly precise coordinates. The mobile device uses GPS* and IMU† sensors in order to properly visualize the object position.

1.1 Previous Work

Vision systems for geo-localization of video image objects are usually based on image registration methods and strongly linked to SLAM and SfM systems. There exists a large amount of literature concerning SLAM and SfM and we refer the reader to the work of Thrun[1] (SLAM) and Triggs[2] (SfM) for introduction.

Depending on the given situation (e.g. GPS-denied areas, type of available background models) this also involves appearance based self-localization and in our case multi-modal image registration methods as well, since our background data involves large scale LiDAR imagery (e.g. Video/LiDAR data registration). Mastin et al.[3] compare synthetically height color coded 2-D renderings with camera images using Mutual information (MI).[4] Vasile et al.[5] derive pseudo-intensity images from LiDAR data including shadows to allow for a 2-D/3-D registration with aerial imagery. Feature based approaches[6,7] mostly rely on the detection and alignment of geometrical features like corners, line segments or planes in the camera image and projections of those from the 3-D data.

Many efforts have been made in the field of augmented reality even for mobile devices with limited hardware resources. For instance Klein and Murray[8] proposed a parallel tracking and mapping (PTAM) framework for augmented reality on mobile phones. Network structures and fixed sensors are used by Yii el al.[9] to create real time augmented reality on smartphones.

Further author information: (Send correspondence to Simon Lemaire)
Simon Lemaire: E-mail: simon.lemaire@fraunhofer.iosb.de

*global positioning system

†inertial measurement unit

Emerging Technologies in Security and Defence; and Quantum Security II; and Unmanned Sensor Systems X, edited by K. L. Lewis, R. C. Hollins, T. J. Merlet, M. T. Gruneisen, M. Dusek, J. G. Rarity, E. M. Carapezza, Proc. of SPIE Vol. 8899, 889919 · © 2013 SPIE · CCC code: 0277-786X/13/$18 · doi: 10.1117/12.2028734

2. METHODS

The workflow of the system starts with the manual selection of an object within an UAV video stream and ends with the visualization of the same object on a mobile device screen. The system can be decomposed into three main building blocks. BB-1: the UAV video stream geo-localization subsystem. Here we use an image based approach for determining highly accurate geo-positions. BB-2: the communication architecture and messaging workflow. Here we transmit the object location to the mobile device. BB-3: the visualization back end on the mobile device. After receiving a highly precise location coordinate we visualize it using augmented reality and traditional map views. The following sections now provide further details about these building blocks.

2.1 UAV Video Stream Geo-Localization

In order to get a highly precise geo-coordinate from UAV imagery we geo-localize the UAV video stream. However, UAV onboard inertial sensors only provide a rough initialization for geo-localization. We assume intrinsically calibrated cameras and determine the 6 DoF camera pose of the UAV sensor data with respect to a geo-referenced background model (e.g. aerial imagery from Google Maps). Based on this information we generate an overlay image, e.g. the exact same camera view using the background data, in order to determine a 1-1 mapping between pixel pairs in the video data and the background model. The calculation of the camera pose is achieved by matching feature correspondences (e.g. SURF Features) between a rendered view of the background model (using the inertial sensors for initialization) and the camera image in combination with an PnP solver.[10] For a detailed description of our registration pipeline we refer to our previous work.[11] In addition to the feature based registration pipeline we additionally perform an intensity based image registration.

2.1.1 Intensity Based Pose Refinement

We additionally maximize an intensity based similarity measure between rendered views and the query images. To achieve very high accuracy for registered poses the convergence range of intensity based multi-modal 2D/3D methods is usually very small. The local feature based pose computation usually provides a sufficiently close starting point. An important design choice is the selection of an appropriate distance measure. Mutual Information[4] is considered the gold standard similarity measure for multi-modal matching. It measures the mutual dependence of the underlying image intensity distributions:

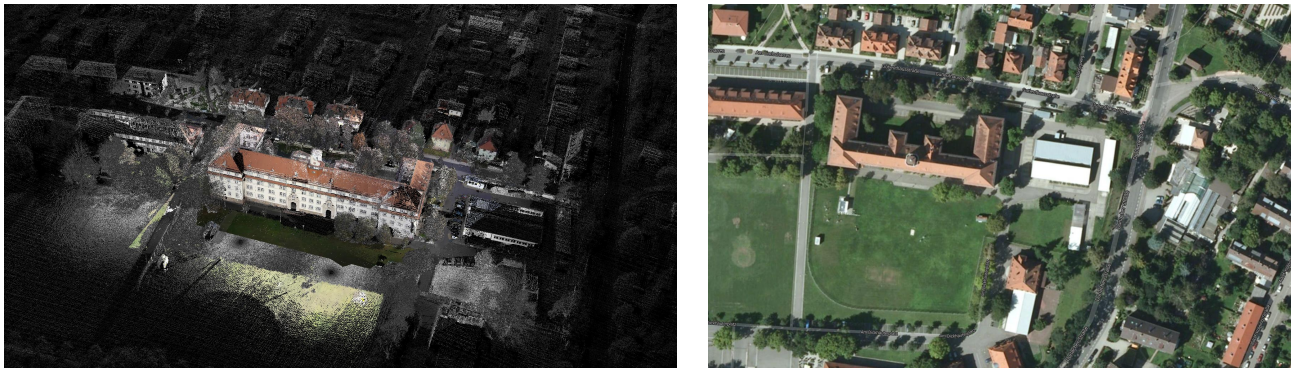
$$D_{(MI)}(I_R, I_{T_\theta}) = H(I_R) + H(I_{T_\theta}) - H(I_R, I_{T_\theta}) \tag{1}$$

where $H(I_R)$ and $H(I_{T_\theta})$ are the marginal entropies and

$$H(I_R, I_{T_\theta}) = \sum_{X \in I_{T_\theta}} \sum_{Y \in I_R} p(X,Y) log(\frac{p(X,Y)}{p(X)p(Y)}) \tag{2}$$

is the joint entropy. $p(X,Y)$ denotes the joint probability distribution function of the image intensities X, Y in I_R and I_{T_θ}, and $p(X)$ and $p(Y)$ are the marginal probability distribution functions. However, MI has usually many local minima near the global optimum. Thus we additionally combine it linearly with a gradient correlation[12] similarity measure for enhancing robustness and accuracy.

The intensity based refinement procedure is usually computationally very expensive. Therefore we use this step for keyframes with the background model and then perform camera tracking based on visual features (FAST). Here we combine a standard recursive Bayesian filter framework for short term tracking and a keyframe based bundle adjustment back-end for dynamical consistency. To achieve real-time performance we only optimize over a window of recently recorded keyframes (windowsize ca. 10-20 imgs) with sufficient camera translation. For a detailed description of the realtime video tracking and registration system we refer to our previous work.[13,14] After geo-referentiation the object coordinates can be transferred to a mobile device whithin the sensor network.



(a) (b)

Figure 1. Background models for image based geo-referentiation - (a) Multiple jointly registered airborne and terrestrial LiDAR scans for low aspect UAV video streams. (b) Orthophoto mosaic representation for high altitude UAV video stream geo-referentiation.

2.2 Information Flow and Communication Architecture

The mobile device is connected with the server via socket connections. Messages are exchanged with a custom protocol based on UDP and images are currently sent as a raw data stream.

The main functionalities are:

- An object is picked in an UAV video stream on the server side. The server sends a XML-message to the mobile device in which the geo-location of the marked object is included.
- The mobile device transmits its position to the server.
- The mobile application transmits the camera image to the server.

2.3 Augmented Reality Visualization on Mobile Devices

The proper visualization of a geo-localized object on a mobile device is still challenging. To this end we visualize the selected geo-localized object using IMU- and GPS-information only.

2.3.1 Visualization using Sensor Data

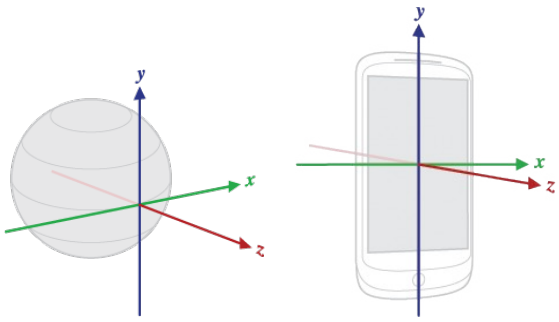
To start the visualization the geo-localized object's position is received from the server as first step. After receiving the object's location the GPS location of the mobile device is determined. The mobile device GPS location is known with an standard GPS error. With this two locations the relative position and distance between them is calculated. The camera field of view is calculated by using the smartphone orientation angles (see 3.1). The smartphone is assumed to be held in portrait position by the user.

As next step a artificial environment is created as described in 3.2. In this environment any augmented reality object can be created.

With the environment set up and the orientation of the mobile device camera known the next step is to visualize the virtual 3D object on the mobile device screen. The positions of the virtual object points on the mobile device screen are derived from the position of the virtual object in the virtual environment. All object points can be included with this projection procedure into the mobile device screen image. This leads to an augmented reality visualization of the selected object on the screen.

2.3.2 Object Visualization using Google Maps

If the object is too far away to be displayed using augmented reality techniques it is favorable to see the own and the object position on a traditional map view. Google Maps offers a good mechanism for realizing such a user interface. The developed visualization can be seen in figure 6.



(a) (b)

Figure 2. Smartphone coordinate axis.

3. IMPLEMENTATION DETAILS

The mobile device system is realized on a Samsung Galaxy Note 2 hardware. The smartphone implementation is based on Android 4.3. The Android application is implemented in JAVA and uses the most recent version (2.4.5) of the Android OpenCV[‡]. Some OpenCV methods use native C++ code with the java native interface (JNI) for performance reasons [§].

3.1 Mobile Device Inertial Measurement Units

The current state of the smartphone is determined from various internal measurement sensors. Android provides the use of several sensors. The proposed application includes the gyroscope sensor (measures the device's rate of rotation), the linear acceleration sensor (measures the acceleration force excluding the force of gravity), the gravitation sensor, the magnetic field sensor and the GPS location sensor. The orientation of the smartphone is obtained by using standard Android techniques. Android uses the gravitation and the magnetic field of the earth to create a global coordinate system shown in figure 2a. The smartphone relative orientation to the global coordinate system is determined by three values:

- Azimuth: rotation around the Z axis.
- Pitch: rotation around the X axis.
- Roll: rotation around the Y axis.

The angles correspond to an Euler angle orientation representation. However, the low IMU sensor quality in custom smartphones results in a large orientation drift and high uncertainties concerning the real orientation. Therefore we implemented a value smoothing on the azimuth, roll and pitch values. Each angle is smoothed separately by taking the last 20 measurements of each angle into account.

1. The arithmetic middle value of these 20 values is calculated.

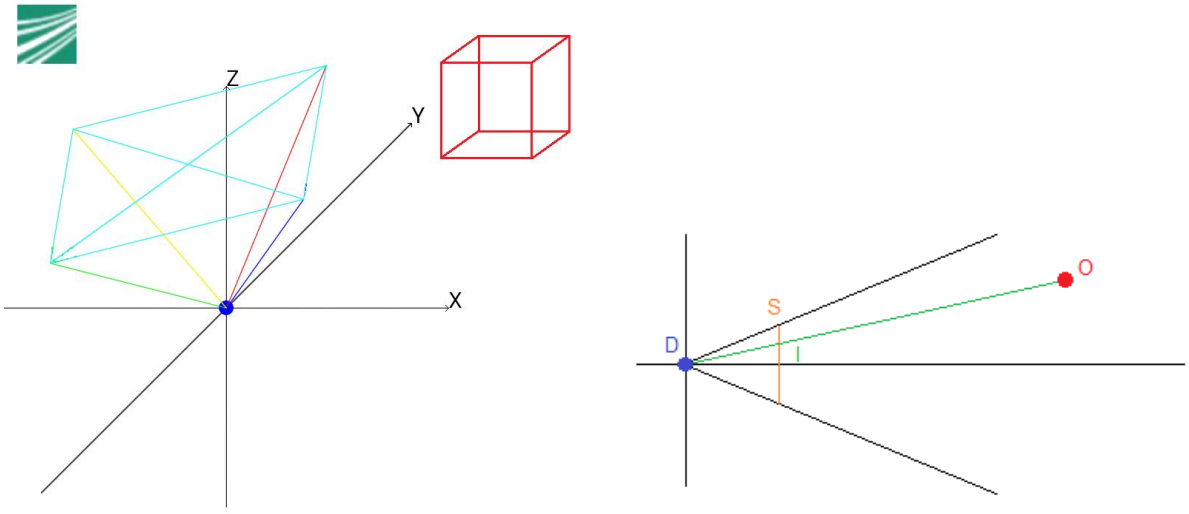
2. Check for each value if the absolute difference between theses value and the arithmetic value is bigger than the maximal authorized deviation.

$$v_j - \left(\frac{1}{N}\sum_{i=1}^{N} v_i\right) < \texttt{deviation limit}; \quad j \in N \tag{3}$$

, with N the number of observed values and j the tested angle value.

[‡]The Android Development Homepage http://developer.android.com/develop/index.html offers a good introduction in the Android development.

[§]More information about the implementation of C++ OpenCV on Android can be found at http://developer.android.com/tools/sdk/ndk/index.html



(a) (b)

Figure 3. (a) virtual coordinate system. The x axis points to the north and the z axis points to the east, which corresponds to the ground of the virtual coordinate system. The z axis corresponds to an altitude. The smartphone camera is situated at the origin of the coordinate system. The blue box describes the field of view of the camera. The augmented reality projection on screen is shown in figure (b).

If all contributing values are within the limits the arithmetic value is taken as current orientation value. Otherwise all values of the three orientation angle are obsolete and no values are returned. This very strict method is used due to the low quality IMU signals. After rotation or acceleration the IMU needs a small time interval to recalibrate. During this period a big drift can appear which is avoided with this test.

3.2 Virtual Environment

As next step a artificial environment system in which any augmented reality object might be displayed is created. This virtual coordinate system is shown in figure 3a, where the x-y plane corresponds to the ground. The z axis is determined by the cross product of x $\times$ y and describes the altitude from the ground. The smartphone camera is situated at the origin of the coordinate system. The light blue box describes the field of view of the camera. The horizontal and vertical view angles of the camera are obtained from the Android platform. In this figure the camera is orientated in North West direction. 3D objects can be included in this environment (shown as red box in this example). The height of the smartphone from the ground will be taken into account in future works.

For the visualization of the server side picked object a virtual 3D object is placed at the same relative position and distance from the mobile device as in real world (e.g. if an object is 20 meters north and 10 meters east from the mobile device it will be placed at the same position in the virtual environment). This procedure is shown as simplified model in figure 3b. The blue dot marks the origin of the coordinate system and the red dot represents an object points. S is a virtual plane which corresponds to the mobile device screen and f is the distance between D and S. A vector from D to O intersects the plane S at the point I. As last step the relative position of the point I on the plane is calculated. This corresponds to the position of the object point on the mobile device screen.

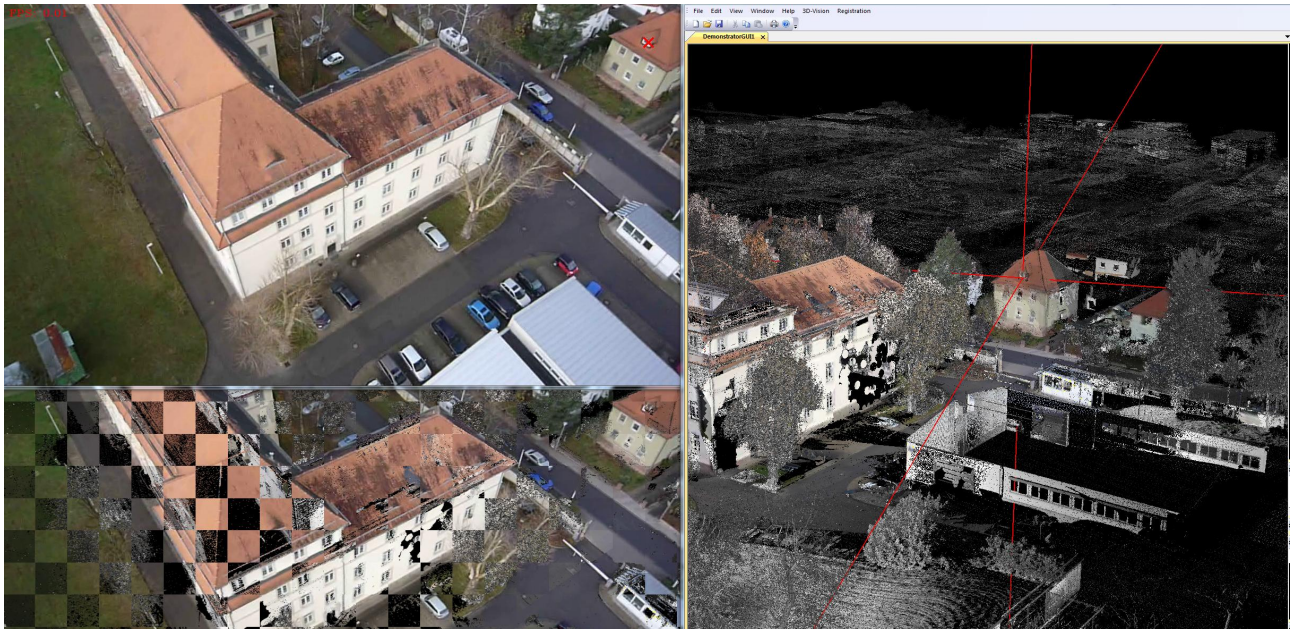


Figure 4. Visualization of the geo-referentiation using low altitude UAV footage and LiDAR data as the underlying background model. Multiple terrestrial laser scans are jointly registered with an aerial laserscan and serve as the underlying 3D model of the environment (bottom-left). Mixed overlay of the camera image with a synthetic LiDAR intensity image generated with the same intrinsic and extrinsic parameters.

4. RESULTS

The feasibility of the approach was experimentally validated using multiple Mini-UAV ("MD-400") videos and one high altitude UAV video in combination with a modern off-the-shelve smartphone (Samsung Galaxy Note 2).

Visualization of the geo-referentiation system are shown in figure 4 and figure 5. The example in figure 4 used low altitude UAV footage and LiDAR data as the underlying background model. Multiple terrestrial laser scans were jointly registered with an aerial laserscan. Figure 5 shows a system which used high altitude UAV footage and orthophotos as the underlying background model.

The proposed video geo-referentiation pipeline operates at video frame rate ($\geq 35FPS$) and achieves a highly accurate registration performance in case of high altitude video footage.

The picked object from the UAV video stream is visualized as augmented reality object on the smartphone camera display. Figure 6a shows the result of the visualization with inertial measurement units where the selected object is represented by a red bar. The visualization precision was improved by using the IMU smoothing method.

Google Maps is used to verify the object position (see figure 6b).

5. CONCLUSION

In this work we presented a system for visualizing objects seen in UAV video feeds on mobile devices. Inertial measurement units and GPS sensors were taken into account. A virtual environment was created to create an augmented reality visualization. Future work will include image matching techniques. These techniques will increase precision and stabilization of the visualization on the mobile device side.

Applications of this approach are widespread and include for instance crisis and disaster management or military applications. To this end we only used an image based geo-localization refinement on the acquisition side. However, we plan to use the 3D-LiDAR background models to enable the proposed registration pipeline on the receiver (Smartphone) side as well. This approach would close the image based registration pipeline loop

resulting in much higher visualization accuracies on the mobile device back-end. Considering the performance and memory size of modern smartphones we are confident to achieve a highly precise geo-localization and visualization of objects seen in smartphone cameras as well.

Figure 5. Visualization of the geo-referentiation system using high altitude UAV footage and Orthophotos as the underlying background model. The left image shows the orthophoto map with the geo-location of the marked pixel (red cross) in the video-stream (upper right image). The transfer is calculated by the registration of the model with the camera image (bottom-right).

REFERENCES

[1] Thrun, S., Burgard, W., and Fox, D., [*Probabilistic Robotics*], MIT Press, Cambridge, MA (2005).

[2] Triggs, B., McLauchlan, P., Hartley, R., and Fitzgibbon, A., [*Bundle Adjustment – A Modern Synthesis*], Springer-Verlag (2000).

[3] Mastin, A., Kepner, J., and Fisher, J., "Automatic registration of lidar and optical images of urban scenes," in [*CVPR*], (2009).

[4] Viola, P. and Wells, W., "Alignment by maximization of mutual information," *International Journal of Computer Vision* **24**(2), 137–154 (1997).

[5] Vasile, A., Waugh, F. R., Greisokh, D., and Heinrichs, R. M., "Automatic alignment of color imagery onto 3d laser radar data," in [*AIPR*], (2006).

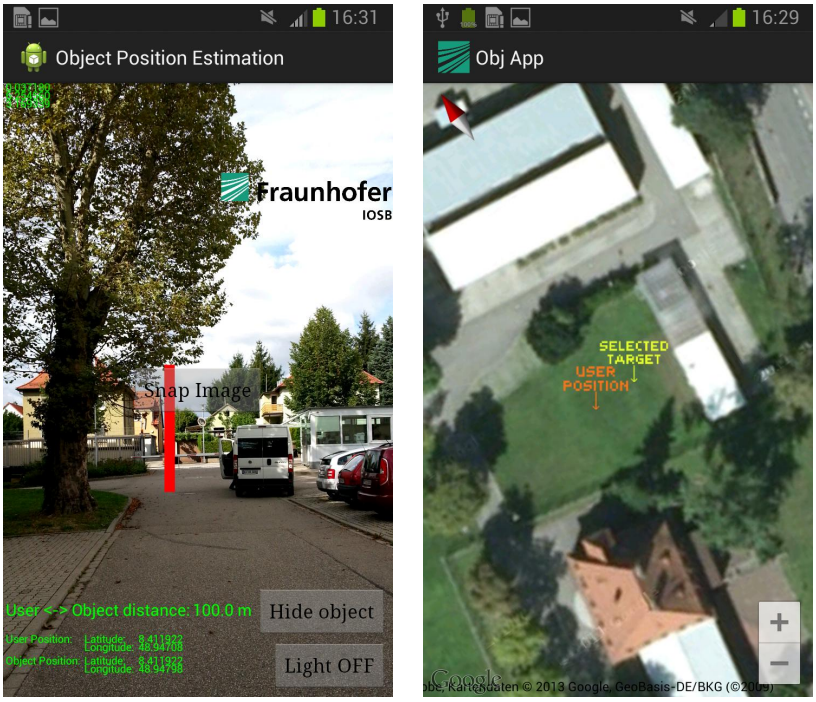
[6] Frueh, C., Sammon, R., and Zakhor, A., "Automated texture mapping of 3d city models with oblique aerial imagery," in [*Proc. 2nd Int. Symp. 3D Data Processing, Visualization and Transmission 3DPVT 2004*], 396–403 (2004).

[7] Ding, M., Lyngbaek, K., and Zakhor, A., "Automatic registration of aerial imagery with untextured 3d lidar models," in [*CVPR*], (2008).

[8] Klein, G. and Murray, D., "Parallel tracking and mapping on a camera phone," *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR07)* (2009).

[9] Yii, W., Li, W. H., and Drummond, T., "Distributed visual processing for augmented reality," *Proc. IEEE International Symposium on Mixed and Augmented Reality (ISMAR12)* (2012).

[10] Lepetit, V., Moreno-Noguer, F., and Fua, P., "Epnp: An accurate o(n) solution to the pnp problem," *International Journal of Computer Vision* **81**, 155–166 (2009).



(a) (b)

Figure 6. (a) shows an augmented reality object obtained by the projection of an virtual object. A selected object is represented by a red bar. A visualization in Google Maps is shown in figure (b)

[11] Bodensteiner, C., Hebel, M., and Arens, M., "Accurate single image multi-modal camera pose estimation," in [*European Conference on Computer Vision (ECCV) - RMLE*], (2010).

[12] Penney, G., Weese, J., Little, J. A., Desmedt, P., Hill, D. L., and Hawkes, D. J., "A comparison of similarity measures for use in 2-d-3-d medical image registration," *IEEE Transactions on Medical Imaging* **17**(4), 586–595 (1998).

[13] Bodensteiner, C. and Arens, M., "Real-time 2d video/3d lidar registration," in [*Proc. 21st Int. Conference on Pattern Recognition (ICPR)*], (2012).

[14] Bodensteiner, C., Hübner, W., Jüngling, K., Solbrig, P., and Arens, M., "Monocular camera trajectory optimization using lidar data," in [*Workshop on Computer Vision in Vehicle Technology - ICCV*], (2011).