Two Cameras and a Screen: How to Calibrate Mobile Devices?

Amaël Delaunoy, Jia Li, Bastien Jacquet and Marc Pollefeys

Department of Computer Science
ETH Zurich, Switzerland

Abstract—We propose a new approach to estimate the geometric extrinsic calibration of all the elements of a smartphone or tablet (such as the screen, the front and the back cameras) by using a planar mirror. By moving a smartphone in front of a single static planar mirror, it is possible to establish correspondences between the images and a pattern displayed on the screen, and therefore estimate the geometric relationship between the non-overlapping cameras with respect to the screen location. The newly proposed setup (static mirror, moving smartphone) enables to both improve the state-of-the-art by working in the minimal case of two images, and improve the accuracy when more images are available. We analyze the minimal case for different calibration scenarios and evaluate the proposed approach on several data. We also show an application of this geometric calibration for specular surface reconstruction, by observing the reflection of a known pattern displayed on the screen.

I. INTRODUCTION

Recent advances in mobile hardware have led to new possibilities for computer vision applications. Any recent smartphone or tablet usually has two cameras (front and back) and a screen. This offers new possibilities for mobile applications exploiting those features. For many applications, the relative position of the screen and the cameras has to be known in advance. This is for example the case for rendering something on the screen that depends on the user’s head position in order to provide more realistic virtual reality feedback, or if one wants to correct for the gaze in video conferencing [1]. Another example would be to use the screen as a reference pattern in order to reconstruct the surface of specular objects. Areas like virtual reality applications, would benefit from having such a calibration algorithm. Those applications, among many others, are based on the assumption that the screen and cameras are already calibrated.

The external calibration of a mobile device consists in finding the relative pose of each of the cameras with respect to the screen, and presents two main challenges. The first one is the estimation of the pose of the screen, which is not directly seen by any of the cameras. The second is the relative poses between front-facing and back-facing cameras which have a non-overlapping field of view. To solve the screen-camera calibration problem, previous works consider the use of a mirror in order to obtain the pose. We improve on this strategy by considering a static mirror instead and by decoupling what is observable both on the moving screen and from the background, pushing the minimal case to two views. Methods for calibrating two non-overlapping cameras, up to our knowledge, need at least three images per camera, as they estimate the per-camera ego-motion and then align it. In our proposed approach, the ego-motion is still implicitly observed by each camera independently, but also through the motion of the screen pattern, reflected on the static planar mirror. This static mirror gives us direct feature correspondences between cameras, enabling the calibration of the camera extrinsics with as few as two views, which therefore improves over the state-of-the-art. This is the main consequence of solving the two calibration problems as a whole (See Figure 1).

A. Related Work

Most previous approaches to solving this problem consider the use of a moving planar mirror. This way, the front facing camera can observe a known reference pattern displayed on the screen. Computing such a pose without a direct view was proposed by Sturm and Bonfort [2]. The idea is to first compute the pose of the virtual camera (reflection of the camera by the mirror plane) using standard resection techniques, and then in a second step, infer the position of the true camera (and the mirror positions). They show how to linearly solve the problem with at least three images, and show that two views leave a one-degree-of-freedom ambiguity. We review this approach later in this paper. Kumar et al. [3] proposed another method that requires five images. Rodrigues et al. [4] proposed later a fast linear approximation based on [2], which allows to accurately recover the pose from at least three images as well. We build on those ideas by also exploiting planar mirrors, but assuming the moving camera see both a static mirror and the background scene enables to solve the problem with less views. A static mirror also allows us to take advantage of the back camera to improve the quality of the screen-camera calibration.

Recently, Agrawal [5] proposed a method that estimates the pose of a camera with respect to a reference object using a single image. A spherical mirror with known radius is required in this case, as well as a known planar reference pattern. While this offers significant advantages, in many practical scenarios, users do not and cannot easily get such a spherical mirror. Alternatively, we propose a flexible, easy-to-use method without the need of a printed pattern. The
reference pattern is directly displayed on the screen of the phone itself, and we only require an common planar mirror, making our approach very practical.

In the above mentioned works, one pose is estimated. In our case, we are interested in calibrating the pose of the back camera as well, which has no overlapping field of view with the front one. Several works have been proposed to compute poses of non-overlapping cameras, in particular in robotic applications, using odometry [6] or SLAM [7]. Those works usually assume that some 3D features can be seen by more than one camera, optionally at different time-steps, permitting the global alignment of each per-camera reconstruction. They can also compute independent per-camera trajectories and align them with each other [8] or using the help of wheel odometry [6]. While this is not a limitation regarding their goal, at least three views are required (i.e. images from all cameras at three different locations or timesteps). In our proposed approach, the ego-motion is implicitly observed by each camera, but also through the motion of the screen pattern reflected on the planar mirror. If the mirror is fixed, the observed reflections gives us direct feature correspondences between cameras, allowing the calibration of all the camera extrinsics.

B. Contributions

In the following, we focus on calibrating a smartphone/tablet (screen and two cameras). The contributions described in this paper can be summarized as:

- We propose a new approach to estimate the complete extrinsic calibration of a mobile device, i.e. the relative poses of each of the front and back cameras with respect to the screen. This approach has a better minimal case of two views (Sect. III), and provides absolute scale thanks to the software-accessible screen dimensions.

- We analyze the mirror based calibration method and show the minimal case falls to two views when the moving smartphone observes both the indirect reflection of its screen on a static planar mirror and some direct projection of 3D world features. (Section III-C)

- We show that considering the problem as a whole enables more accurate calibration than independent calibration of each parts (screen-camera and camera-camera). This is quantitatively demonstrated on both synthetic and real data in IV-B, and qualitatively through an application to specular surface reconstruction in Section IV-D.

II. PROBLEM FORMULATION

A. Acquisition Scenario

As described above, in order to calibrate the phone, we consider the case where the phone takes synchronized images from both sides in front of a planar mirror. The screen displays a known reference pattern, referred in this paper as screen pattern. The world describes everything that is not rigidly connected to the phone and that is considered to be static (rigid) in this paper. Please note that this could be either an other reference pattern not attached to the phone, or natural features of the 3D world. The front camera sees the world through the planar reflection as well as the reflection of the screen pattern. The back camera sees the world with a direct view. By moving the phone around, images from both cameras are taken simultaneously, and used to calibrate the phone, i.e. the relative poses of each of the cameras with respect to the screen pattern. Note that in this work, we consider the intrinsic parameters to be known in advance, but their estimation (or at least their refinement) could also be included in the process. Figure 1 illustrates an acquisition example.

B. Models and Notations

A phone is represented by its screen, its front-facing camera, and its back camera, all modeled by standard pinhole camera models. A 3D point X in space is projected into the camera following the projection x ~ K [R | t] X, R and t being the rotation and translation respectively, and K the camera intrinsics. The pose of the phone at time instant i, denoted by V_i, can be defined as V_i = \begin{bmatrix} R_i & t_i \\ 0 & 1 \end{bmatrix}.

Let \Pi_i = (n_i^T d_i)^T be the mirror plane parameters associated with the phone pose i, and parameterized by its normal n_i^T and distance to origin d_i. The planar reflection S_i can be written S_i = \begin{bmatrix} I_3 - 2n_i n_i^T & 2d_i n_i \\ 0 & 1 \end{bmatrix} in the global coordinate system. A virtual camera is the symmetric of a camera through a reflection S_i. A phone dual camera \( (V_i, V^F, V^B) \) is a combination of a front camera \( V^F \) and a back camera \( V^B \), in the phone coordinate system \( V_i \). In the following we denote by view a pair of images taken from both cameras simultaneously.
III. FULL EXTRINSIC CALIBRATION OF SMARTPHONES

A. Screen, Front and Back Cameras: Geometric Calibration

Let’s consider a configuration where we have \( n \) poses of the smartphone \( (\mathcal{V}_i)_{1 \leq i \leq n} \) in the world coordinate system. The back camera observes the scene directly, therefore its pose can be written as \( \mathcal{V}_i^B = \mathcal{V}_i^B \mathcal{V}_i \). At the same time, the front camera observes both the scene and the pattern on screen through the mirror reflection. The poses of the virtual front camera in the world coordinate system are \( \mathcal{V}_i^F \mathcal{S}_i = \mathcal{V}_i^F \mathcal{S}_i \mathcal{V}_i \) and we have \( \mathcal{V}_i^F = \mathcal{V}_i^F \mathcal{S}_i \mathcal{V}_i^{-1} \) in the phone coordinate system. We want to estimate the front and back camera poses \( (\mathcal{V}_i^F, \mathcal{V}_i^B) \) with respect to the screen position, given the following measurements:

1) \( (\mathcal{V}_i^F \mathcal{S}_i, \mathcal{V}_i^F)_{1 \leq i \leq n} \), the poses of the virtual front cameras and back cameras in the world coordinate system. They can be estimated via a classic Structure from Motion algorithm, using the world screen features.

2) \( (\mathcal{V}_i)_{1 \leq i \leq n} \), the poses of the virtual front cameras in the phone coordinate system.

1) Initial solution of \( \mathcal{V}_i^F \): In the smartphone coordinate system, we can rewrite the virtual camera poses \( \mathcal{V}_i^F \) as:

\[
\mathcal{V}_i^F = \mathcal{V}_i^F \mathcal{S}_i \, .
\]

where \( \mathcal{S}_i = \mathcal{V}_i \mathcal{S}_i \mathcal{V}_i^{-1} \) are the mirror reflections in the smartphone coordinate system. The goal is to recover the real camera pose \( \mathcal{V}_i^F \) and the mirror reflections \( \mathcal{S}_i \) given the virtual poses \( (\mathcal{V}_i^F)_{1 \leq i \leq n} \). Here, we recognize an IPMR (image of a planar mirror reflection). In order to solve that problem and estimate the real camera poses, [2], [3], [4] already proposed different approaches. We use the approach described in Rodrigues et al. [4], which appears to be efficient. The idea is to replace \( \mathcal{V}_i^F = \mathcal{V}_i^F \mathcal{S}_i \mathcal{V}_i^{-1} \) in (1) to eliminate \( \mathcal{V}_i^F \).

2) Initial solution of \( \mathcal{V}_i^B \): One should notice that an SfM algorithm only estimates \( (\mathcal{V}_i^F \mathcal{S}_i, \mathcal{V}_i)_{1 \leq i \leq n} \) up to a scale factor. Namely, there exists a unique scale factor \( s \), by which the estimation of the camera poses by SfM differ from the real poses:

\[
\begin{cases}
\mathcal{V}_i^F \mathcal{S}_i = \Lambda_s \mathcal{V}_i^F \Lambda_s^{-1} \\
\mathcal{V}_i^B = \Lambda_s \mathcal{V}_i^B \Lambda_s^{-1} 
\end{cases}
\]

where \( \Lambda_s = \begin{pmatrix} s & 0 \\ 0 & 1 \end{pmatrix} \).

In order to recover \( \mathcal{V}_i^B \), we are going to explore the relation between \( (\mathcal{V}_i^F, \mathcal{V}_i^B) \) and \( (\mathcal{V}_i^F) \). First of all, we compute the pose of a virtual front camera related to its corresponding back camera. We denote this pose by \( \mathcal{U}_i^F \), and we obtain:

\[
\mathcal{U}_i^F = \mathcal{V}_i^F (\mathcal{V}_i^B)^{-1} = \Lambda_s^{-1} \mathcal{V}_i^F \mathcal{S}_i (\mathcal{V}_i^B)^{-1} \Lambda_s 
\]

This equation tells us that there exists a rigid transformation and a scale linking the pose of the virtual front camera in the phone coordinate system to the one in the back camera coordinate system. Thus, the pose of the back camera \( \mathcal{V}_i^B \) and \( s \) are such that:

\[
\forall \, i \in \{1, \ldots, n\}, \quad \mathcal{V}_i = \Lambda_s \mathcal{U}_i^F \Lambda_s^{-1} \mathcal{V}_i^B 
\]

We could formulate this as a least squares problem in order to find an initialization. \( \mathcal{V}_i^F \) and \( s \) are the solutions of the following optimization problem:

\[
\arg \min_{\mathcal{W} \in \mathbb{S}O(3), \, s \in \mathbb{R}} \sum_{i=1}^{n} \left| \mathcal{V}_i^F - \Lambda_s \mathcal{U}_i^F \Lambda_s^{-1} \mathcal{W} \right|^2 .
\]

Let us compute the analytical solution of this optimization problem. In fact, \( \mathcal{W}, \mathcal{V}_i, \mathcal{U}_i^F \) and \( \mathcal{U}_i^B \) are elements of \( \mathbb{S}E(3) \), so can be written as:

\[
\mathcal{W} = \begin{pmatrix} R_w & T_w \\ 0 & 1 \end{pmatrix}, \quad \mathcal{V}_i^F = \begin{pmatrix} R_i^F & T_i^F \\ 0 & 1 \end{pmatrix}, \quad \mathcal{U}_i^F = \begin{pmatrix} R_i^U & T_i^U \\ 0 & 1 \end{pmatrix}.
\]

By replacing in each term of Equation (5), we have:

\[
\left| \mathcal{V}_i^F - \Lambda_s^{-1} \mathcal{U}_i^F \Lambda_s \mathcal{W} \right|^2 = \left| R_i^F - R_i^F R_i^U \right|^2 + \left| T_i^F - s T_i^F - R_i^U T_i^U \right|^2 .
\]

In this way, the problem (5) is equivalent to the two following subproblems:

\[
\begin{align*}
\arg \min_{R^U \in \mathbb{S}O(3)} & \sum_{i=1}^{n} \left| R_i^F - R_i^U R_i^U \right|^2 , \\
\arg \min_{T^U \in \mathbb{R}^3} & \sum_{i=1}^{n} \left| T_i^F - s T_i^F - R_i^U T_i^U \right|^2 .
\end{align*}
\]

The problem in Equation (8) is an orthogonal Procrustes problem, finding the best orthogonal matrix approximation, whose solution can be computed analytically; Equation (9), on the other hand, is a quadratic form without constraints.
and its solution can be found by vanishing the first order derivative. The final result of the optimization becomes:

\[
\begin{align*}
W &= \begin{pmatrix} G F^T & J - s H \\ 0 & 1 \end{pmatrix}, \\
S &= \left( \sum_{i=1}^{n} T_{\text{B}_{i}} \right) \left( \sum_{i=1}^{n} T_{\text{B}_{i}} T_{\text{B}_{i}} \right)^{-1},
\end{align*}
\]

where

\[
\begin{align*}
J &= \frac{1}{n} \sum_{i=1}^{n} R_{i}^T T_{i}^v, \quad H = \frac{1}{n} \sum_{i=1}^{n} R_{i}^T T_{i}^w, \\
A_{i} &= R_{i}^T T_{i}^v - J, \quad B_{i} = R_{i}^T T_{i}^w - H,
\end{align*}
\]

and F and G are such that \( \sum_{i=1}^{n} R_{i}^T R_{i} = F \), D = G is the SVD of \( \sum_{i=1}^{n} R_{i}^T R_{i} \).

\section{B. Bundle Adjustment}

Once we have obtained an initial guess of the phone calibration, we can refine the solution in a global bundle adjustment problem. We refine the 3D points \( X \) of the camera, positions, and back \( V^w \) and front cameras \( V^F \), by minimizing the reprojection error between the projection of 3D points \( X \) and their corresponding previously detected 2D feature positions. This is similar to standard bundle adjustment (BA) problems [9], except that, in this case, the front cameras have different constraints. The error residuals can be decomposed into three block parameters:

- Reprojection error between the projection of the 3D points of the 3D world in the back camera and their corresponding 3D features extracted using Structure-from-Motion. This is a classical BA problem [9]. We denote the set of those “world-back” points by \( \mathcal{N}_{bw} \).
- Reprojection error between the projection of the 3D points of the 3D world in the front image through the planar reflection, and their corresponding 2D features from the structure-from-motion. This gives a link between the front and the back camera, with absolute scale, and is the key component of our bundle adjustment. We denote the set of those “front-world” points by \( \mathcal{N}_{fw} \).
- Reprojection error between the 3D points of the reference screen pattern and their correspondences in the front image through the planar reflection. This is typically what is done to refine the solution in [4], [2]. We denote the set of those “front-screen” points by \( \mathcal{N}_{fs} \).

In the end, we can minimize the following energy function:

\[
E(X, V_i, V^F, V^I) = \sum_{j \in \mathcal{N}_{bw}} f_j(X, V_i, V^I)^2 + \sum_{j \in \mathcal{N}_{fw}} f_j(X, V_i, V^F, V^I)^2 + \sum_{j \in \mathcal{N}_{fs}} f_j(V_i, V^F, V^I)^2,
\]

where the functions \( f_j \) are standard reprojection errors on the image plane.

\section{C. Fixed Mirror Case}

In the previous section, there was no constraint on the mirror positions. In this section we consider the case where the planar mirror is fixed with respect to the 3D world. In this case, we can constrain the solution and reduce the number of parameters by estimating only one mirror position in the bundle adjustment, which is a straightforward extension of Section III-B.

In this scenario, we can directly estimate the pose of the mirror with respect to the camera without the need of the back camera. There just need to be enough features to estimate the virtual camera poses. While an optimized solution via bundle adjustment is often preferred in the end to obtain more accurate results, some applications could benefit from a linear solution. In the following, we show that we can get a solution with a minimal case of two images in this context.

\textbf{Minimal Case: The Two Views Case:}

Using the fixed mirror scenario, we are interested in analyzing the Screen - Front-facing camera calibration case. With the planar mirror, the front camera can observe a known pattern displayed on the screen through the planar reflection. This is very similar to [2], [4], but instead we consider the case where two reference scenes are available. In fact, the virtual front camera not only observes the screen, but also the 3D world. So in this case, we have two references, one fixed to the phone (the screen), and another one (a printed pattern or simply the 3D world). If one keeps the phone fixed and moves mirrors, then both the screen and the 3D world are the same reference. In this case [2], [4] applies and we need at least three images (three mirror positions). Instead, we consider the two reference object case and two phone positions (and images) and show it is sufficient to recover the pose of the camera with respect to the screen.

When only two images with two different positions are available \((n = 2)\), the solution has been shown to be ambiguous [2]. A set of solutions around a fixed-axis rotation are all solutions to the problem (see Figure 3 for details). This happens because the 3D world and reference object are considered together. In our setup, two “reference” objects are available, and this is sufficient to disambiguate the calibration.
A necessary and sufficient condition is:
\[ \det(M_1 - M_2, v_1, v_2) = 0 . \] (12)
It results that:
\[ \det(M_1 - M_2, u_1, u_2) - s \det(e, v_1, v_2) = 0 . \] (13)
Equation (13) is a necessary condition to find a solution to our problem. It means that we will find a solution when \( \det(e, v_1, v_2) \neq 0 \). In particular, pure translations can not be handled in our case, but this is the same limitation than in prior works [2], [3], [4] and for any planar mirror based calibration.

It is worth noting that, if we are provided two views containing both front and back images, we can get the screen-front camera calibration and directly register the back camera using the correspondences with the reflected scene observed in the front image.

D. Algorithm Summary - Complete Approach

Our approach to estimate the complete geometric calibration of a mobile device can be summarized as the following:
1) We compute the relative poses of the front camera with respect to the screen in the phone coordinate system using Rodrigues et al. [4].
2) We then obtain an initial estimate of the back camera poses as well as the virtual front camera poses, all in the world coordinate system using structure-from-motion (or using a reference pattern located in the world if available, which was not used in this paper).
3) We compute an initial estimation of the phone camera poses using the approach described in Section III, and in particular Equation (10). It allows us to bring all cameras in the same world coordinate system with absolute scale (or up to a single scale factor if the size of the screen is unknown).
4) We refine the initial solution of the camera poses, the phone poses, the mirror positions and the 3D points in a global bundle adjustment minimizing the energy in Equation (11). (Section III-B)

IV. Experiments

A. Setup and Implementation

In practice, the calibration of the phone is performed by taking pictures at the same time of both front and back images. Due to a lack of APIs and mobile devices, it is currently not possible to get an easy access to devices that offer the capability to save synchronized images simultaneously. To be able to calibrate it anyway, we leave the phone fixed for a second to switch the data acquisition between the front and the back camera. We believe this technical issue will not be a problem in the future as last generation phones tend to include this feature and there will soon be publicly available APIs.

All the implementation has been written in C++. The Structure-from-Motion used is the one in [10], [11] where we use the feature extraction, matching and initial pose estimation via triplets. In order to solve the bundle adjustment problem, our implementation uses the non-linear Ceres Solver [12] with the three parameter block functions described in Section III-B. While the StFm part currently
only works off-line on a PC, the rest runs on both PC and Android phones.

B. Results

We generated several simulated datasets (10) with varying mirror and camera positions, and for each dataset we added a Gaussian noise on the 2D measurements and run the algorithm 100 times. The obtained mean error values are displayed in Figure 5 and represent the error in \( \text{mm} \) of the camera position with respect to the screen. The results are shown for 4 different standard deviations of noise level. For each of the graphs the error is shown with dependency on the number of images used.

As expected, the full bundle adjustment significantly improves the calibration accuracy of the phone, and the front camera in particular. The use of the back camera brings more information for the estimation of the calibration of the front camera with respect to the screen. Our approach is both more accurate and robust than the Front-Camera-vs-Screen calibration only (referred to as Front/Camera BA in Figure 5). As a comparison, in Figure 5 we use the linear solution of [2], [3], [4] on top of which we run a non-linear bundle adjustment (please note that after bundle adjustment, those three methods all lead to the same result).

We also applied our calibration to real phone data, namely the "Wiko Cink Peax 2" and the "Samsung Galaxy S4". Results are shown in Figure 6, and the result of the calibration relative to the screen are (translations are in \( \text{mm} \)):

\[
V_{CP}^F = \begin{pmatrix} -0.9999 & 0.0036 & 0.001 & -5.848 \\ 0.0035668 & 0.9999 & -0.0081 & -49.617 \\ -0.00104 & -0.0081 & -0.9999 & 9.7046 \end{pmatrix},
\]

\[
V_{CP}^B = \begin{pmatrix} 0.9996 & 0.00272 & -0.0259 & 2.7134 \\ -0.00267 & 0.9999 & 0.00177 & -28.60 \\ 0.0259 & -0.0017 & 0.999 & 1.7067 \end{pmatrix},
\]

\[
V_{S4}^F = \begin{pmatrix} -0.9995 & -0.0017 & -0.0319 & -8.76868 \\ -0.0015 & 0.9999 & -0.00913 & -9.0586 \\ 0.03199 & -0.00908 & -0.99944 & -0.3332 \end{pmatrix},
\]

\[
V_{S4}^B = \begin{pmatrix} 0.9999 & 0.00422 & 0.0097 & 1.4274 \\ -0.00423 & 0.9999 & 0.00175 & -29.933 \\ -0.00972 & -0.00179 & 0.9999 & -4.1557 \end{pmatrix}.
\]

Those computations correlate with our measurements on the devices (less than 1 \( \text{mm} \)). In the Galaxy S4 dataset Figure 6, the mirror was fixed. However, our calibration result do not use this constraint and estimate the position of the mirror for each phone position. The reconstructed planes are all well aligned as shown in Figure 6. We use this constraint as a validation, where we find a standard deviation between all the estimated mirror planes of 0.4\( \text{mm} \) in distance along the z axis and 0.04 degrees in angle.

C. Front-Camera-vs-Screen Minimal Case

In the case of the Front-Camera-vs-Screen calibration, we evaluate our method on synthetic data and compare it with Rodrigues et al. [4]. The simulated data is obtained the same way as in the previous section. Here we only need two images in our case, and three images in the compared case. Data are averaged by randomly sampling 50 pairs (our method) or triplets (Rodrigues et al.) of cameras and repeating the experiment 30 times for each dataset. The

![Figure 6. Phone calibration on real data. Left: Visualization of the screen, front and back camera positions on a CAD model of the Galaxy S4. Right: Sparse 3D reconstruction and final calibration estimation with estimated mirror positions. Virtual cameras with respect to the mirror planes (in the middle) are shown using corresponding colors.](attachment:image1.png)

![Figure 7. Calibration Results - Visualization of the reprojection error, before (Left) and after Bundle Adjustment (Right). Pattern points are reprojected as green dots, the front camera center as a cyan dot, and the back camera center as a red dot.](attachment:image2.png)

![Figure 8. Left: Comparison with Rodrigues et al. [4] in the minimal case of the Front camera / Screen calibration. [4] uses three images when our approach uses only two. For small noise level, our approach is more robust as it takes advantage of the world reference. Right: Result visualization on the Wiko dataset seen in the camera coordinate system.](attachment:image3.png)
results are illustrated in Figure 8.

It is worth to notice that our approach improves the accuracy of the calibration even though we only used two images. This is made possible because we take advantage of a second reference object in the world coordinate system.

D. Application Case: Multi-View Specular Surface Reconstruction

Our calibration approach can be used to reconstruct specular surfaces. The idea is to use the front camera to observe the reflection of a reference pattern shown on the screen, then apply a single view reconstruction method to estimate the specular surface. At the same time, the back camera observing the world helps to locate the front camera poses, which allows us to combine the single view result to a multi-view reconstruction. We reconstruct the specular surface as follows:

1) Inspired by the feature-based deformation surface tracking of Bartoli [13], we estimate a warp transformation between the observed pattern and the reference pattern in the screen. In our case, SIFT [14] is used in order to get a robust matching between the distorted reflected pattern and the original one.

2) For each independent view we estimate the specular surface via a similar approach to [15]

3) We compute SfM using the back camera, and use the calibration approach described in this paper to find the positions of all the front cameras. All the cameras being in the same world coordinate system, we can combine all the independently reconstructed surfaces.

E. Discussion

Our approach takes advantage of both cameras and we demonstrated that the expected improvement in accuracy is indeed obtained. Based on the same ideas as previous works, our solution also does not work with pure translational motions, but we found this to hardly happen in practice. Agrawal [5] address this issue at the cost of requiring a spherical mirror. This change of requirement also leads to a calibration of both cameras with a single image per camera.

Agrawal [5] indeed obtained. Based on the same ideas as previous works, our solution also does not work with pure translational motions, but we found this to hardly happen in practice. However, the calibration procedure becomes impractical as such mirrors are not so common and not easily available.

One possible future direction is being able to run everything on the phone itself, by using, for example, the SLAM algorithm of [16] on both cameras simultaneously.
Scanning specular surfaces with a mobile device look very promising. A dedicated reconstruction algorithm integrating all the multi-view information is however necessary to obtain a coherent surface.

V. CONCLUSION

We proposed a new practical approach to calibrate a set of two non-overlapping cameras rigidly connected to a reference pattern. In a case of a Smartphone or Tablet, it corresponds to finding the front and back camera poses relative to its screen. By exploiting the fact they are rigidly connected, we can calibrate all the element of the device with good accuracy and absolute scale with the use of a simple planar mirror. Considering the calibration problem as a whole, we improved over the minimal case of three views for each subproblem to two views for the problem as a whole, when using a static mirror. This novel algorithm allows the use of the back facing camera to improve the robustness and accuracy of the front-screen calibration. We have demonstrated the accuracy of our calibration both quantitatively on synthetic and real data, and qualitatively on a specular surface reconstruction application. Our calibration setup offers an approach which is more practical and more robust than previous ones, opening doors to many 3D mobile vision applications that require a complete phone calibration.

ACKNOWLEDGMENT

This work was supported by the Swiss National Science Foundation under the project Nr. 156973.

REFERENCES


