Photometric Patch-based Visual-Inertial Odometry

Xing Zheng, Zack Moratto, Mingyang Li and Anastasios I. Mourikis

Abstract— **In this paper we present a novel direct visual-inertial odometry algorithm, for estimating motion in unknown environments. The algorithm utilizes image patches extracted around image features, and formulates measurement residuals in the image intensity space directly. One key characteristic of the proposed method is that it models the true irradiance at each pixel as a random variable to be estimated and marginalized out. The formulation of the photometric residual explicitly accounts for the camera response function and lens vignetting (which can be calibrated in advance), as well as unknown illumination gains and biases, which are estimated on a per-feature or per-image basis. We present a detailed evaluation of our algorithm on 50 datasets with high-precision ground truth, which amount to approximately 1.5 hours of localization data. Through a direct comparison with a point-feature based method, we demonstrate that the use of photometric residuals results in increased pose estimation accuracy, with approximately 23% lower estimation errors, on average.**

I. INTRODUCTION

The ability to accurately estimate the 3D position and orientation of a device in a GPS-denied environment is essential in several applications such as robotics, augmented reality, and virtual reality. Because of the complementary information provided by an inertial measurement unit (IMU) and a camera, the combination of these two sensors for 3D localization has attracted considerable research interest. Among the key challenges that researchers have tried to address is the fact that cameras naturally produce high-dimensional measurements (e.g., in the order of 10^5 pixels per image). To allow for real-time processing, we must be able to exploit the most useful localization information in the images, while keeping the computational cost low.

The "traditional" way of achieving this is by detecting point features in the images (such as SIFT [1], FAST [2], or Shi-Tomasi corners [3]). Typically, only up to a few hundred features are used in each image – a significant reduction in dimensionality compared to the original image size. While the use of point features greatly decreases the number of measurements that need to be processed in the estimator, it also suffers from a number of drawbacks. First, it results in discarding information from the unused areas in the image. Second, the varying levels of "distinctiveness" of each point feature, which may translate to varying levels of measurement accuracy, are typically not modelled. Third, point-feature extraction may fail altogether in fast-motion or low-light situations, leading to a complete failure of the estimator. Finally, it should be noted that feature extraction and matching may itself be a time-consuming process.

Xing Zheng and Anastasios Mourikis are with the Department of Electrical and Computer Engineering, University of California, Riverside. Zack Moratto and Mingyang Li are with Google Inc.

To avoid these shortcomings of feature-based methods, there has been renewed interest in so-called "direct methods", which directly employ the image-intensity measurements in the localization algorithm (see, e.g. [4]–[6] and references therein). While, in theory, direct approaches could allow using the measurements of every pixel in an image, and naturally model the local distinctiveness of each image area, they also suffer from shortcomings. The "photometric" (i.e., image-intensity) measurements are sensitive to changes in the camera exposure time and gains, lighting conditions, camera viewing angles, surface properties, and other factors. This can make it difficult to model the relationship between the intensity of the projection of the same scene point in different images. While prior work has offered evidence that direct approaches can lead to improved performance over feature-based ones, the comparisons in the existing literature have generally involved very different systems. This makes it difficult to tease out the effects of using a feature-based vs. a direct approach under the same conditions.

In this paper, we describe a new approach for directly using the intensity measurements in distinctive image patches for localization. A key characteristic of the proposed approach is that it models the true irradiance at each pixel as an unknown random variable. Since estimating this random variable is not our primary interest, it is marginalized out during the formulation of the measurement residual. Additionally, in our approach we employ a detailed radiometric camera model that accounts for gamma correction and lens vignetting. In this work, we do not employ the photo-consistency assumption commonly used in direct approaches, and instead model an illumination gain and bias as random variables, to be estimated in our measurement model. Taken together, the above characteristics allow us to accurately model the uncertainty in the image-intensity measurements and their correlation among frames, resulting in increased estimation precision.

The proposed direct measurement model can be applied with several possible estimator formulations (e.g., extended Kalman filter (EKF), sliding window iterative minimization). We here choose to employ this model in conjunction with a sliding-window EKF estimator for visual-inertial odometry, the multi-state-constraint Kalman filter (MSCKF) 2.0 [7], [8]. A key goal of this paper is to allow for a direct comparison between the "traditional" point-feature-based approach and the photometric one. To this end, we select the image patches to be used in the photometric formulation around the *same* feature points used in the point-based MSCKF. We perform extensive testing using 50 datasets recorded under varying conditions, each with high-precision ground truth

provided by a Vicon system. The results demonstrate that the photometric approach yields, on average, higher localization accuracy, reducing the average position errors by 23%.

II. Related Work

Prior work in the area of visual-inertial localization is extensive, and providing a full review within the limited space available is impossible. We here discuss the most relevant approaches, with respect to four different criteria:

Measurement type: Most existing approaches for visual-inertial localization are feature-based ones. The vast majority of these approaches employ point features, but lines have also been used [9], [10]. In contrast to such methods, we here focus on algorithms that directly use image intensities for forming a measurement model. Depending on the type of image regions used in the algorithm, these can be further divided into dense methods, where the entire image is used [11], semi-dense methods, where only regions with large gradient magnitude are used [12], [13], and patch-based methods, where regions around extracted point-features are used [5], [14], [15]. Our approach belongs to the last category.

Camera models: In the feature-based formulation, only a camera's geometric model [16], [17] is generally considered. By contrast, direct approaches also need to model the image formation process, i.e., the mapping from light irradiance to image intensities. A simple, commonly-used model for direct approaches assumes that the measured image intensity at a given pixel is proportional to the irradiance of the incoming light. In practice, however, the camera usually has a nonlinear response function and suffers from lens attenuation. The calibration of these two effects has been considered in [18], [19], and we here also employ a similarly calibrated camera. As shown in [6] (which is a vision-only formulation), modeling these effects can improve estimation performance.

Feature models: For localization in an unknown environment, feature-based approaches generally model the 3D positions of the features as random variables, either to be estimated along with the pose states [20], or to be marginalized to impose constraints on pose states [7], [21], or a combination of both [22]. Similar considerations apply to direct methods. In our approach, feature states are modeled as random variables and marginalized out in the update, thus allowing a probabilistically correct use of the features' information.

In addition to the feature positions, direct approaches must also deal with the *appearance* of the features (or of the collection of pixels considered). Often, this is not modeled in a probabilistic formulation, and instead it is assumed that the appearance (image intensity) of corresponding pixels is the same between images [4], [23]. This assumption, often termed the photo-consistency constraint, is a strong one, and can be violated by changes in the exposure time, scene illumination, or camera viewing angle. A less constraining model is to assume that the *irradiance* is the same between images (the so-called irradiance-consistency assumption). This is done, for instance, in [6]. However, even this approach does not properly model the fact that the actual irradiance is a random variable, that needs to be estimated along with the feature position and possibly other variables. In our work, the irradiance, as well as the illumination gain and bias, are all modeled as random variables in the measurement model.

Estimator choice: Most feature-based visual-inertial algorithms are either Kalman-filter-based methods [7], [20], [24], or methods employing iterative minimization [25]–[27]. On the other hand, most direct approaches are formulated as energy minimization problems, and solved by iterative algorithms [12], [28]. Only few EKF-based direct approaches have been proposed to date [14], [29]. Both of these algorithms employ an EKF state vector that includes the feature positions (in [29] these are represented in a robocentric map), and use the photo-consistency constraint in order to obtain measurement residuals. By contrast, the method we propose here does not include the feature positions in the EKF state vector, which has certain computational advantages, as explained in [7]. Moreover, we formulate a more expressive modified irradiance-consistency constraint, which is able to better model the imaging mechanism.

III. Filter Formulation

We now describe the proposed algorithm for visual-inertial localization, which is based on the sliding-window formulation of the MSCKF 2.0 algorithm [7], [8]. Specifically, the state vector of the estimator contains the M poses where the last M images were recorded, while observations of scene features are employed for imposing constraints between these poses. In the original, point-feature-based formulation, these constraints were derived using the image coordinates of the features' projections in the images. By contrast, in the new formulation presented here, the constraints are derived by directly using the image intensity measurements in a patch around each detected feature.

In the remainder of this section we briefly describe the formulation of the state vector, as well as the propagation and state management of the MSCKF algorithm, while the photometric update is described in detail in Section IV.

A. Formulation

We consider a platform equipped with an IMU and a monocular grayscale global-shutter camera, moving in an area populated with naturally-occurring features, whose coordinates are not known a priori. Our goal is to estimate the position and orientation of the platform with respect to a gravity-aligned global coordinate frame, $\{G\}$, using the inertial measurements and the camera images. To derive the estimator's equations, we affix a coordinate frame $\{I\}$ to the IMU, and a coordinate frame $\{C\}$ to the camera. We here assume that the camera is intrinsically calibrated, and the frame transformation between $\{I\}$ and $\{C\}$ is known.

The IMU state at time-step k is described by the vector:

$$\mathbf{x}_{I_k} = \begin{bmatrix} {}^{I_k}_{G}\bar{\mathbf{q}}^T & {}^{G}\mathbf{p}_k^T & {}^{G}\mathbf{v}_k^T & \mathbf{b}_{\mathbf{g}_k}^T & \mathbf{b}_{\mathbf{a}_k}^T \end{bmatrix}^T \quad (1)$$

where[1] ${}^{I_k}_{G}\bar{\mathbf{q}}$ is the unit quaternion [30] representing the rotation from the global frame $\{G\}$ to the IMU frame $\{I\}$ at time-step k, ${}^{G}\mathbf{p}_k$ and ${}^{G}\mathbf{v}_k$ are the IMU position and velocity in the global frame, and $\mathbf{b}_{\mathbf{g}_k}$ and $\mathbf{b}_{\mathbf{a}_k}$ are the gyroscope and accelerometer biases, respectively, which are modeled as Gaussian random-walk processes.

The IMU error-state is defined as:

$$\tilde{\mathbf{x}}_{I_k} = \begin{bmatrix} {}^{G}\tilde{\boldsymbol{\theta}}_k^T & {}^{G}\tilde{\mathbf{p}}_k^T & {}^{G}\tilde{\mathbf{v}}_k^T & \tilde{\mathbf{b}}_{\mathbf{g}_k}^T & \tilde{\mathbf{b}}_{\mathbf{a}_k}^T \end{bmatrix}^T \tag{2}$$

where the standard additive error definition is used for the position, velocity and biases (i.e., for a random variable $\mathbf{y}$, its estimate is denoted $\hat{\mathbf{y}}$, and the estimation error is defined as $\tilde{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}$), while for the orientation errors we use a minimal 3-dimensional representation, as defined in [8].

The estimator state vector contains the current IMU state, and M states corresponding to the latest M images:

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{x}_{I_k}^T & \boldsymbol{\pi}_{k-M}^T & \boldsymbol{\pi}_{k-M+1}^T & \cdots & \boldsymbol{\pi}_{k-1}^T \end{bmatrix}^T \tag{3}$$

where each of the states $\boldsymbol{\pi}_\ell$, $\ell = k-M, \ldots, k-1$ consists of the IMU pose at the time the ℓ-th image was recorded, as well as the "illumination parameter" $\boldsymbol{\eta}_\ell$ of the corresponding image (see Section IV-D):

$$\boldsymbol{\pi}_\ell = \begin{bmatrix} \mathbf{x}_{\mathbf{p}_\ell}^T & \boldsymbol{\eta}_\ell^T \end{bmatrix}^T, \quad \text{with} \quad \mathbf{x}_{\mathbf{p}_\ell} = \begin{bmatrix} {}^{I_\ell}_{G}\bar{\mathbf{q}}^T & {}^{G}\mathbf{p}_\ell^T \end{bmatrix}^T \tag{4}$$

B. Propagation and State Augmentation

Every time an IMU measurement is received, it is used to propagate the IMU state and covariance matrix, as described in [8]. Similarly to the original MSCKF, when an image is recorded, a copy of the current IMU pose and the corresponding illumination parameter are inserted into the state vector (3). Specifically, if a new image is recorded at time-step $k+1$, we augment the state vector (3) with the IMU-state estimates:

$$\hat{\boldsymbol{\pi}}_{k+1|k} = \begin{bmatrix} \hat{\mathbf{x}}_{\mathbf{p}_{k+1|k}}^T & \hat{\boldsymbol{\eta}}_{k+1}^T \end{bmatrix}^T \tag{5}$$

where the initialization of $\boldsymbol{\eta}_{k+1}$ is described in Section IV-D. The filter's covariance matrix is also augmented as follows:

$$\mathbf{P}_{k+1|k} \leftarrow \begin{bmatrix} \mathbf{P}_{k+1|k} & \mathbf{P}_{k+1|k}\mathbf{J}_{\mathbf{p}}^T & \mathbf{0} \\ \mathbf{J}_{\mathbf{p}}\mathbf{P}_{k+1|k} & \mathbf{J}_{\mathbf{p}}\mathbf{P}_{k+1|k}\mathbf{J}_{\mathbf{p}}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_{\boldsymbol{\eta}_0} \end{bmatrix} \tag{6}$$

where $\mathbf{J}_{\mathbf{p}}$ is the Jacobian of the IMU pose $\mathbf{x}_{\mathbf{p}_{k+1}}$ with respect to the state vector, and $\mathbf{P}_{\boldsymbol{\eta}_0}$ is the initial covariance of the illumination parameter $\boldsymbol{\eta}_{k+1}$. Once the state augmentation is complete, the image is processed so that an EKF update is performed.

[1]Notation: The preceding superscript for vectors (e.g., X in ${}^{X}\mathbf{a}$) denotes the frame of reference with respect to which quantities are expressed. ${}^{X}_{Y}\mathbf{R}$ is the rotation matrix rotating vectors from $\{Y\}$ to $\{X\}$, and ${}^{X}_{Y}\bar{\mathbf{q}}$ is the corresponding unit quaternion. ${}^{X}\mathbf{p}_Y$ is the origin of frame $\{Y\}$ with respect to $\{X\}$. $\mathbf{0}$ and $\mathbf{I}$ are the zero and identity matrices respectively, while $\hat{a}$ and $\tilde{a}$ represent the estimate, and error of the estimate, of a variable a, respectively.

IV. EKF Update

In the MSCKF approach, each feature is being tracked through multiple images. Once the feature is lost, or its track length reaches the length of the sliding window, M, all the feature's observations are used at the same time for an EKF update. We follow the same approach in the proposed photometric formulation of the MSCKF as well. The difference lies in the fact that, while in the original MSCKF the measurement residuals are defined in the space of image coordinates (i.e., the residuals are the feature reprojection errors), in the photometric formulation the residuals are defined in the space of image intensities. Specifically, for each feature we define a planar patch centered around the feature 3D position, and consider the projection of this patch in each of the images. The measurement residuals are defined by enforcing a modified irradiance-consistency assumption among all images.

In what follows, we describe the geometric model of the camera used in our experiments, our radiometric model that includes the camera response function (gamma correction) and lens vignetting, and finally the formulation of the residuals used for the EKF update.

A. Geometric Model

Consider a point with global 3D position ${}^{G}\mathbf{p}$. The image coordinates of the point's projection in the camera at time step ℓ will be a function of the position vector ${}^{G}\mathbf{p}$, the IMU pose $\mathbf{x}_{\mathbf{p}_\ell}$, and the camera projection geometry. While our approach is applicable with any camera geometry, we here describe the model of [16] that we employ for the fisheye camera used in our experimental setup. With this model, the image projection coordinates are given by:

$$\mathbf{h}\left({}^{G}\mathbf{p}, \mathbf{x}_{\mathbf{p}_\ell}\right) = \frac{1}{r_u\omega}\arctan\left(2r_u\tan\left(\frac{\omega}{2}\right)\right)\begin{bmatrix} a_u u \\ a_v v \end{bmatrix} + \mathbf{p}_c \tag{7}$$

where $\mathbf{p}_c$ is the pixel location of the principal point, (a_u, a_v) are the camera focal length measured in horizontal and vertical pixel units, ω is the distortion parameter, and

$$r_u = \sqrt{u^2 + v^2} \tag{8}$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{{}^{C_\ell}z}\begin{bmatrix} {}^{C_\ell}x \\ {}^{C_\ell}y \end{bmatrix} \tag{9}$$

$$\begin{bmatrix} {}^{C_\ell}x \\ {}^{C_\ell}y \\ {}^{C_\ell}z \end{bmatrix} = {}^{C}_{I}\mathbf{R}\ {}^{I_\ell}_{G}\mathbf{R}({}^{G}\mathbf{p} - {}^{G}\mathbf{p}_\ell) + {}^{C}\mathbf{p}_I \tag{10}$$

In the last equation, ${}^{C}_{I}\mathbf{R}$ is the rotation matrix from the IMU to the camera frame, and ${}^{C}\mathbf{p}_I$ is the position of the origin of $\{I\}$ in the camera frame. We here assume that the camera is both intrinsically and extrinsically calibrated, and therefore the parameters $\mathbf{p}_c, a_u, a_v, \omega$, ${}^{C}_{I}\mathbf{R}$ and ${}^{C}\mathbf{p}_I$ are known.

B. Radiometric Model

In an "ideal" camera, the measured image intensity at a given pixel $\mathbf{p}$ would be proportional to the irradiance of the

incoming light at the given pixel:

$$I_{\text{ideal}}(\mathbf{p}) = a\xi(\mathbf{p}) \tag{11}$$

where $\xi(\mathbf{p})$ is the light irradiance, and a is a scaling parameter that accounts for the physical size of the pixel on the sensor, as well as the image exposure time. However, in practice, cameras generally have a nonlinear response function to the incoming light energy (the so-called gamma correction), and lenses cause attenuation of the incoming light, which is typically more pronounced towards the edges of the image (so-called vignetting). Therefore, the measured intensity in a real camera can be modeled as:

$$I_o(\mathbf{p}) = F(V(\mathbf{p})a\xi(\mathbf{p})) + n_p \tag{12}$$

where the function $F(\cdot)$ represents the camera response function, $V(\mathbf{p})$ is the lens attenuation at pixel $\mathbf{p}$, and n_p is additive observation noise (e.g., electronic noise).

The camera response function, F, can be estimated via calibration, by taking pictures of a constant scene with several known exposure settings, and creating a lookup table (see, e.g., [19]), or fitting the data with the standard gamma-correction model:

$$F(x) = cx^{\gamma} \tag{13}$$

where c and γ are constants to be estimated via fitting. Similarly, the lens attenuation function can be estimated via calibration, e.g., by obtaining images of a uniformly diffused plane. Once these parameters are known, we can obtain the "rectified" image intensity for each pixel in an image:

$$I(\mathbf{p}) = \frac{F^{-1}(I_o(\mathbf{p}))}{V(\mathbf{p})} \tag{14}$$

where F^{-1} is the inverse function of F. This rectified intensity value is related to the light irradiance by:

$$I(\mathbf{p}) = a\xi(\mathbf{p}) + n \tag{15}$$

where n is additive image-intensity noise. In the remainder of the paper, the intensity measurements will always refer to the rectified intensity, unless otherwise stated.

In the system used in our experiments, the value of γ is known by design, which removes the need for a calibration of the camera response function. Moreover, we have performed a calibration of the lens attenuation function using the radially-symmetric vignetting model in [18]. The resulting function V is shown in Fig. 1.

C. Modified irradiance-consistency constraint

We now discuss the formulation of the modified irradiance-consistency constraint, which forms the basis for computing the EKF residuals. Our goal is to derive an equation that relates the observed image intensities at corresponding pixel locations across multiple images. Specifically, let us consider a point feature, f_i, observed in the M images of the sliding window. Our formulation uses the assumption that the scene structure around this feature is *locally* well-modeled by a planar patch P_i. For simplicity, and similarly to [5], in this work we define the normal vector of P_i to be

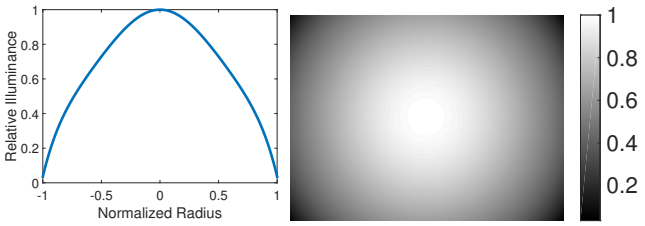

Fig. 1. Vignetting calibration results. Left: the radially-symmetric vignetting function computed for our camera. The x axis represents the distance from the image center. Right: visualization of the vignetting effect on the image.

parallel to the optical ray from the feature to the camera, at the time one of the images was recorded (we term this image the "anchor" image, and select it to be the first image where f_i was seen). The direction of the normal vector is treated as a known constant in our work, but it could also be estimated (and marginalized out later on), if desired.

In our formulation, photometric residuals are computed by considering the image projections of a set of points on P_i. Specifically, we begin by defining N^2 image locations on an $N \times N$ pixel grid centered at the projection of f_i in the anchor image. By "back-projecting" these image coordinates to P_i, we obtain N^2 3D points, with positions ${}^G\mathbf{p}_{i,j}$, $j = 1, \ldots, N^2$. It is important to point out that the selection of the N^2 points in the anchor image is done in a deterministic way, in an $N \times N$ grid around the observed feature coordinates. Therefore, the only random variables that are involved in determining the 3D positions ${}^G\mathbf{p}_{i,j}$ are (i) the IMU pose at the time the anchor image was recorded, $\mathbf{x}_{\mathbf{p}_A}$, and (ii) the distance of P_i to the camera at the time the anchor image was recorded. In our work we parameterize this distance by its inverse, ρ_i. In what follows we use the notation ${}^G\mathbf{p}_{i,j} = {}^G\mathbf{p}(\rho_i, \mathbf{x}_{\mathbf{p}_A}, j)$ to express the fact that ${}^G\mathbf{p}_{i,j}$ is a function of ρ_i, $\mathbf{x}_{\mathbf{p}_A}$, and known quantities.

We now proceed to obtain relationships for the image intensity at the projections of the N^2 patch points on all the M images. From (15), we obtain:

$$I_\ell(\mathbf{h}({}^G\mathbf{p}(\rho_i, \mathbf{x}_{\mathbf{p}_A}, j), \mathbf{x}_{\mathbf{p}_\ell})) = a_\ell \xi_{i,j,\ell} + n_{i,j,\ell} \tag{16}$$

where $I_\ell(\mathbf{h}({}^G\mathbf{p}(\rho_i, \mathbf{x}_{\mathbf{p}_A}, j), \mathbf{x}_{\mathbf{p}_\ell}))$ is the (rectified) image intensity at the projection of the point ${}^G\mathbf{p}_{i,j}$ in the ℓ-th image, and $\xi_{i,j,\ell}$ is the light irradiance produced by this point in the ℓ-th camera frame. If the surface being imaged was perfectly Lambertian, and the global lighting conditions remained the same, then the irradiance values across all images (i.e., for all $\ell \in \{k-M, \ldots, k-1\}$), would be the same. In practice, however, this is not the case, and the irradiance may change slightly among images. We model this as:

$$\xi_{i,j,\ell} = \alpha_{i\ell}\xi_{i,j} + \beta_{i\ell} \qquad j = 1, \ldots, N^2 \tag{17}$$

In other words, we assume that the irradiance of a patch in different images is related by an (unknown) linear function.

From (16) and (17), and by combining the measurements from all the N^2 points belonging to the patch, we obtain:

$$\mathbf{I}_\ell(\rho_i, \mathbf{x}_{\mathbf{p}_A}, \mathbf{x}_{\mathbf{p}_\ell}) = a_{i\ell}\boldsymbol{\xi}_i + b_{i\ell}\mathbf{1} + \mathbf{n}_{i\ell} \tag{18}$$

where we have defined $a_{i\ell} = a_\ell \alpha_{i\ell}$ and $b_{i\ell} = a_\ell \beta_{i\ell}$, $\mathbf{1}$ is an $N^2 \times 1$ vector of ones, $\mathbf{I}_\ell(\rho_i, \mathbf{x}_{\mathbf{p}_A}, \mathbf{x}_{\mathbf{p}_\ell})$ is the vector containing the intensity values at the projections of all N^2 patch points in image ℓ, i.e., a vector with elements:

$$[\mathbf{I}_\ell(\rho_i, \mathbf{x}_{\mathbf{p}_A}, \mathbf{x}_{\mathbf{p}_\ell})]_j = I_\ell(\mathbf{h}({}^G\mathbf{p}(\rho_i, \mathbf{x}_{\mathbf{p}_A}, j), \mathbf{x}_{\mathbf{p}_\ell})),\ j=1,...,N^2,$$

the vector $\boldsymbol{\xi}_i$ is defined as:

$$\boldsymbol{\xi}_i = \begin{bmatrix}\xi_{i,1} & \xi_{i,2} & \cdots & \xi_{i,N^2}\end{bmatrix}^T \tag{19}$$

and finally $\mathbf{n}_{i\ell}$ is the noise vector, modeled as zero-mean, white, Gaussian, with covariance matrix $\sigma^2\mathbf{I}_{N^2}$.

The equation in (18) is the modified irradiance-consistency constraint we employ in this work. It relates the image intensities at the projection coordinates of the N^2 patch points to the irradiance of the patch, as well as the illumination parameters $a_{i\ell}$ and $b_{i\ell}$. Since the projection coordinates are functions of the IMU poses and the patch's inverse depth, this constraint provides the necessary connection between the geometric and photometric quantities. Note that the patch irradiance and the illumination parameters are unknown random variables, which have to be either marginalized out, or included in the state vector of the filter and estimated. The exact process we follow for this is detailed in the following section, which describes the way the constraint in (18) is used for formulating EKF residuals.

D. Photometric MSCKF Update

As explained earlier, at the time when a feature is lost from tracking, or its feature track length reaches the size of the sliding window, all its measurements are used for an EKF update. The process for feature f_i begins by using the feature's projections in the images to obtain an estimate of the feature's inverse depth, ρ_i, via least-squares minimization, as in the original MSCKF algorithm. Using this estimate, as well as the estimate for the IMU poses (available from the MSCKF state vector), we can compute the projection coordinates of the N^2 patch points in all M images, and the image intensities observed at these locations, $\mathbf{I}_\ell(\hat{\rho}_i, \hat{\mathbf{x}}_{\mathbf{p}_A}, \hat{\mathbf{x}}_{\mathbf{p}_\ell})$, $\ell = k-M, \ldots, k-1$. Moreover, an estimate for the "illumination gain" parameter, $a_{i\ell}$, can be obtained as $\hat{a}_{i\ell} = t_\ell / t_o$, where t_ℓ is the exposure time of image ℓ, and t_o is a nominal minimum exposure time. An initial estimate for the "illumination bias" parameter can be chosen simply as $\hat{b}_{i\ell} = 0$. Finally, the irradiance vector $\boldsymbol{\xi}_i$ can be estimated as $\hat{\boldsymbol{\xi}}_i = (\mathbf{I}_A(\hat{\rho}_i, \hat{\mathbf{x}}_{\mathbf{p}_A}, \hat{\mathbf{x}}_{\mathbf{p}_A}) - \hat{b}_{iA})/\hat{a}_{iA}$. Using these estimates, we can compute the following measurement residuals:

$$\mathbf{r}_{i\ell} \doteq \mathbf{I}_\ell(\hat{\rho}_i, \hat{\mathbf{x}}_{\mathbf{p}_A}, \hat{\mathbf{x}}_{\mathbf{p}_\ell}) - \hat{a}_{i\ell}\hat{\boldsymbol{\xi}}_i - \hat{b}_{i\ell}\mathbf{1}, \tag{20}$$

for $\ell = k-M, \ldots, k-1$. All these residuals can be stacked in a block vector $\mathbf{r}_i$, whose block elements are $\mathbf{r}_{i\ell}$:

$$\mathbf{r}_i = [\mathbf{r}_{i\,k-M}^T \quad \cdots \quad \mathbf{r}_{i\,k-2}^T \quad \mathbf{r}_{i\,k-1}^T]^T \tag{21}$$

This residual vector uses all intensity measurements corresponding to the patch of feature i, across all images. To use it in an EKF update, we must also compute the Jacobian matrices that relate the residual to the estimation errors. To this end, we begin by linearizing (20), which yields:

$$\begin{aligned}\mathbf{r}_{i\ell} \approx\ & \hat{a}_{i\ell}\tilde{\boldsymbol{\xi}}_i + \hat{\boldsymbol{\xi}}_i\tilde{a}_{i\ell} + \tilde{b}_{i\ell}\mathbf{1} - \mathbf{H}_{\rho_{i\ell}}\tilde{\rho}_i \\ & - \mathbf{H}_{\mathbf{p}_{i\ell}}\tilde{\mathbf{x}}_{\mathbf{p}_\ell} - \mathbf{H}_{\mathbf{p}_{iA}}\tilde{\mathbf{x}}_{\mathbf{p}_A} + \mathbf{n}_{i\ell}\end{aligned} \tag{22}$$

where $\mathbf{H}_{\rho_{i\ell}}$, $\mathbf{H}_{\mathbf{p}_{i\ell}}$, and $\mathbf{H}_{\mathbf{p}_{iA}}$ are the Jacobians of the observed image intensities with respect to the feature inverse depth, the IMU pose at time step ℓ and the anchor pose, respectively. Specifically, $\mathbf{H}_{\rho_{i\ell}}$, $\mathbf{H}_{\mathbf{p}_{i\ell}}$, and $\mathbf{H}_{\mathbf{p}_{iA}}$ are block matrices with N^2 rows, with the j-th row given by:

$$\begin{aligned}\mathbf{H}_{\rho_{i\ell,j}} &= \nabla I_\ell(\mathbf{h}({}^G\mathbf{p}(\hat{\rho}_i, \hat{\mathbf{x}}_{\mathbf{p}_A}, j), \hat{\mathbf{x}}_{\mathbf{p}_\ell}))^T \frac{\partial \mathbf{h}}{\partial {}^G\mathbf{p}_{i,j}} \frac{\partial {}^G\mathbf{p}_{i,j}}{\partial \rho_i} \\ \mathbf{H}_{\mathbf{p}_{i\ell,j}} &= \nabla I_\ell(\mathbf{h}({}^G\mathbf{p}(\hat{\rho}_i, \hat{\mathbf{x}}_{\mathbf{p}_A}, j), \hat{\mathbf{x}}_{\mathbf{p}_\ell}))^T \frac{\partial \mathbf{h}}{\partial \mathbf{x}_{\mathbf{p}_\ell}} \\ \mathbf{H}_{\mathbf{p}_{iA,j}} &= \nabla I_\ell(\mathbf{h}({}^G\mathbf{p}(\hat{\rho}_i, \hat{\mathbf{x}}_{\mathbf{p}_A}, j), \hat{\mathbf{x}}_{\mathbf{p}_\ell}))^T \frac{\partial \mathbf{h}}{\partial {}^G\mathbf{p}_{i,j}} \frac{\partial {}^G\mathbf{p}_{i,j}}{\partial \mathbf{x}_{\mathbf{p}_A}}\end{aligned}$$

where ∇I_ℓ is the image gradient function for image ℓ. We note that the above Jacobians are computed using the first-available estimates of each IMU position, to ensure filter consistency [8]. For the same reason, we do not directly use the image gradient computed from differencing the ℓ-th image in the above calculations. Instead, ∇I_ℓ is computed by transforming ∇I_A to the ℓ-th image.

The residual defined in (20) and its linearized approximation in (22) involve not only the IMU poses that we are interested in estimating, but also the feature inverse depth, feature-patch irradiance, and the illumination parameters. These are effectively "nuisance parameters," which we can proceed to marginalize, similarly to what is done in the original MSCKF with the feature position. With respect to the illumination gain and bias however, we can explore an additional option: instead of estimating and then marginalizing these parameters "locally," on a per-feature, per-image basis, we can treat one or both of these parameters as being constant for all features within an image. This approach allows us to model "global" effects, such as changes in the entire scene's illumination. To employ this global approach, the parameters can be included in the state vector of the MSCKF, and estimated on a per-image basis (see (4)).

In Section V-C all four options for modeling the illumination gain and bias as either local or global are explored. We here present the case where the illumination bias is modeled as a global parameter, and thus $\boldsymbol{\eta}_\ell = b_{i\ell}$ is included in the state vector (3) (and the same value is used for all features in image ℓ), while the illumination gain is modeled as a local parameter. Thus, we can rewrite (22) as:

$$\mathbf{r}_{i\ell} \approx \mathbf{H}_{\mathbf{x}_{i\ell}}\tilde{\mathbf{x}}_k + \mathbf{H}_{\mathbf{y}_{i\ell}}\tilde{\mathbf{y}}_i + \mathbf{n}_{i\ell} \tag{23}$$

where $\tilde{\mathbf{x}}_k$ is the filter error-state vector at time-step k, $\tilde{\mathbf{y}}_i$ contains the error-states to be marginalized for feature i:

$$\tilde{\mathbf{y}}_i = [\tilde{\boldsymbol{\xi}}_i^T \quad \tilde{a}_{i\,k-M} \quad \cdots \quad \tilde{a}_{i\,k-1} \quad \tilde{\rho}_i]^T$$

and the Jacobians $\mathbf{H}_{\mathbf{x}_{i\ell}}$ and $\mathbf{H}_{\mathbf{y}_{i\ell}}$ are given by:

$$\begin{aligned}\mathbf{H}_{\mathbf{x}_{i\ell}} &= \begin{bmatrix}\mathbf{0} & \cdots & [-\mathbf{H}_{\mathbf{p}_{iA}} & \mathbf{0}] & \cdots & [-\mathbf{H}_{\mathbf{p}_{i\ell}} & \mathbf{1}] & \cdots & \mathbf{0}\end{bmatrix} \\ \mathbf{H}_{\mathbf{y}_{i\ell}} &= \begin{bmatrix}\hat{a}_{i\ell}\mathbf{I}_{N^2} & \hat{\boldsymbol{\xi}}_i\mathbf{e}_\ell^T & -\mathbf{H}_{\rho_{i\ell}}\end{bmatrix}\end{aligned}$$

where $\mathbf{e}_\ell$ is the $(\ell - k + M + 1)$-th canonical basis vector of dimension M. Using (23), and stacking these expressions for all the residuals $\mathbf{r}_{i\ell}$, $\ell = k - M, \ldots, k - 1$, we obtain

$$\mathbf{r}_i \approx \mathbf{H}_{\mathbf{x}_i} \tilde{\mathbf{x}}_k + \mathbf{H}_{\mathbf{y}_i} \tilde{\mathbf{y}}_i + \mathbf{n}_i \tag{24}$$

where $\mathbf{H}_{\mathbf{x}_i}$ and $\mathbf{H}_{\mathbf{y}_i}$ are matrices with block rows $\mathbf{H}_{\mathbf{x}_{i\ell}}$, and $\mathbf{H}_{\mathbf{y}_{i\ell}}$, respectively, and $\mathbf{n}_i$ a block vector with elements $\mathbf{n}_{i\ell}$.

The expression in (24) is the linearized approximation of (21) that we sought to obtain. However, since this expression contains both the error of the EKF state vector, $\tilde{\mathbf{x}}_k$, as well as the error vector $\tilde{\mathbf{y}}_i$, which does not involve states in the EKF state, we proceed to compute a new residual that does not include the latter. This is done by a process similar to that used in the original MSCKF. Specifically, we define a matrix $\mathbf{V}_i$ whose columns form a basis for the left nullspace of $\mathbf{H}_{\mathbf{y}_i}$, and define a new residual $\mathbf{r}_i^o$ as:

$$\mathbf{r}_i^o = \mathbf{V}_i^T \mathbf{r}_i \simeq \mathbf{H}_i^o \tilde{\mathbf{x}}_k + \mathbf{n}_i^o \tag{25}$$

where $\mathbf{H}_i^o = \mathbf{V}_i^T \mathbf{H}_{\mathbf{x}_i}$ and $\mathbf{n}_i^o = \mathbf{V}_i^T \mathbf{n}_i$. For computational efficiency, we can compute $\mathbf{r}_i^o$ and $\mathbf{H}_i^o$ without explicitly computing $\mathbf{V}_i$ [7]. Once $\mathbf{r}_i^o$ and $\mathbf{H}_i^o$ are computed, we proceed by performing a Mahalanobis gating test to reject outliers, and patches whose residuals pass the test are employed in an EKF update, analogously to [7]. Once the update to the state estimate and the covariance matrix are computed, the oldest camera state is removed from the state vector, to maintain a sliding window of bounded length.

V. Experimental validation

In this section we present the results of experimental testing that was carried out to compare the photometric formulation to the original, point-based formulation of the MSCKF, and to examine the effect of a number of parameters on algorithm performance. For this testing, we used a collection of 50 datasets with high-quality ground truth, thus allowing for a thorough performance evaluation. In these experiments, a Project Tango developer tablet was held by a person moving in a room monitored by a Vicon motion-capture system. The duration of each recorded dataset is between 1 and 2 minutes, and sample images from the datasets are shown in Fig. 2. A variety of motions were generated, including walking, sudden stopping, running, and fast rotations, to enable evaluation in a wide range of conditions.

During the experiments, an exterior rigid frame with reflective markers was attached to the tablet. The Vicon system provides 500-Hz sub-millimeter accuracy motion-tracking estimates of four markers on the exterior frame. To obtain the transformation between the exterior and the IMU frame, "hand-eye calibration" has been performed offline [31], using the Vicon estimates for the exterior frame and the IMU-trajectory estimates computed via full visual-inertial bundle adjustment. With the calibrated transformation, the position estimates of the markers can then be used to provide the ground-truth estimates for the IMU frame.

In order to isolate the effects of using photometric residuals (as opposed to residuals defined as re-projection errors), in our implementation both the patch-based photometric approach and the original, point-feature MSCKF, employ the same feature tracking and matching processes. The same feature tracks are used by all algorithms in the experiments. Moreover, the same point-feature-based triangulation is used to provide the initial guess of the point-feature or patch positions in all cases (note that, if desired, the triangulation could also be formulated based on the photometric residuals).

Fig. 2. Sample images recorded during the experiments.

Since RANSAC is used for outlier rejection in our feature-matching process, different feature tracks will, in general, be generated from the same dataset with different random seeds, and slightly different results will be generated by the filter. Therefore, the result from a single run of an estimator on a given dataset may not be sufficiently representative of the algorithm's performance. To address this issue, we process each dataset with each algorithm 10 times, with a different random seed each time (all the compared algorithms use the same set of random seeds, for a fair comparison). From these results we compute the following performance metrics:

- Typical Error: After processing a dataset with one algorithm, we compute the root-mean-square error (RMSE) of the position estimates over the entire trajectory. Subsequently, we compute the median of these RMSEs over the 10 times the algorithm is run on the same dataset with a different random seed. After repeating this process for all 50 datasets, we compute the average of the 50 results. This metric represents the "expected performance" of the algorithm.
- 90th-percentile error: After processing a dataset with one algorithm, we compute the 90-th percentile of the position error norm throughout the entire dataset. Subsequently, we compute the 90-th percentile of these values over the 10 times the algorithm is run on the same dataset with a different random seed. After repeating this process for all 50 datasets, we compute the average of the 50 results. This metric represents what we expect the (close to) worst-case performance to be.

A. Patch-based vs. Point-feature-based

We first compare the performance of the proposed patch-based formulation against that of the original, point-feature-based MSCKF formulation. For this test, the illumination bias is treated as a "global" parameter in each image, while the illumination gain is treated as a "local" parameter for each feature in each image (as presented in Section IV-D). Table I lists the two performance metrics we are interested

TABLE I

PATCH-BASED VS. POINT-FEATURE-BASED FORMULATION

	Typical Error (m)	90-th Percentile Error (m)
Point-based	0.176	0.270
Patch-based	0.135	0.208

in for the two approaches. We can observe that the patch-based approach achieves lower errors than the original point-feature-based approach. Specifically, both the typical error and the 90-th percentile error decrease by approximately 23%. In 66% of the 50 datasets, the patch-based approach achieves errors smaller than the original point-feature-based one. While these reductions may not appear dramatic, they are significant, since the starting point of the comparison (the point-based MSCKF) is already heavily tuned and optimized for accuracy.

These results demonstrate the potential of the direct approach to improve estimation performance. To our knowledge, this is the first time this result has been observed in a setting where the compared algorithms *only* differ in the way in which the residuals are formulated. In previous comparisons that have appeared in the literature, the compared systems typically have significant differences beyond the use of a photometric vs. geometric residual, which makes it difficult to attribute the observed differences in performance to a specific factor.

B. Effect of Camera Model Fidelity

We now turn our attention to examining the effects of different parameters and design choices on the performance of the direct photometric formulation. In the following experiments, we only employ the patch-based algorithm, but in each experiment we change one aspect of the model while keeping the others fixed, to evaluate the effects of the change.

We start by comparing the effects of using camera models with different levels of detail. In Table II, we present the results of the proposed, "full model," compared to the cases where (i) lens vignetting is not modelled, or (ii) the per-pixel irradiance of the patch is not treated as a random variable, and is instead assumed to be equal to the observed irradiance in the first image. We can see that in both cases, the estimation accuracy is reduced. The loss of performance when vignetting is not modeled is expected, as the camera used exhibits significant vignetting. Prior work has also pointed to the importance of using a detailed model for lens vignetting [18]. However, we note that the need to model the irradiance as an unknown quantity to be estimated is usually ignored in prior photometric approaches. Typically, the irradiance is computed from the intensity measurements at the reference frame, and then treated as a true value (as done for the test in case (ii) here). However, this causes unmodeled errors, because the measurement at the reference frame contains noise that is not accounted for. More importantly, the photometric residuals computed with this irradiance estimate are correlated with each other, but this correlation is not modeled. In our approach, the irradiance is treated as a random variable and thus its estimation error as well as the correlations between the photometric residuals are properly accounted for. This leads to improved performance, as seen in Table II.

TABLE II

EFFECT OF CAMERA MODEL FIDELITY

	Typical Error (m)	90-th Percentile Error (m)
Full Model	0.135	0.208
No Vignetting	0.139	0.214
No Irradiance	0.148	0.230

TABLE III

ILLUMINATION PARAMETERS: LOCAL VS. GLOBAL

Illumination Gain	Illumination Bias	Typical Error (m)	90-th Percentile Error (m)
Global	Global	0.151	0.239
Global	Local	0.137	0.212
Local	Global	0.135	0.208
Local	Local	0.149	0.235

C. Illumination Parameters: Local vs. Global

As discussed in Section IV-D, we can either model the illumination parameters (gain and bias) as global ones to be included in the state vector for each image, or as local ones, to be marginalized when processing each feature's measurements. In Table III, we present results showing the performance of all four possible combinations. As we can see, while all cases outperform the point-based formulation, the use of a "global" illumination bias and "local" illumination gain outperforms all other options. This indicates that the unmodeled changes in illumination (e.g., due to a flickering light source or exposure-time changes) are better modeled as global parameters for all features in an image, while the effects of non-Lambertian surfaces, which cause changes in irradiance by camera viewpoint, are better modeled as a per-feature, per-image gain.

D. Performance with different patch sizes

We evaluated the performance of the algorithm when varying the size of the patches defined around each feature, and the results are shown in Table IV. We can observe that the filter's estimation accuracy increases at first, and then decreases as the size of the patch grows. We attribute this to the fact that using a larger patch leads to more information being used (more accurate localization of the features), and this initially leads to improved accuracy. However, when the size of the patch is large, the assumption of the scene being locally planar does not hold anymore, and thus the patch measurements are more likely to be rejected in the Mahalanobis test, resulting in worse performance.

VI. CONCLUSION

In this paper, we have presented a direct formulation of the MSCKF algorithm for visual-inertial odometry. The algorithm utilizes image patches extracted around image feature positions, and formulates measurement residuals in

TABLE IV

PERFORMANCE WITH DIFFERENT PATCH SIZES

Patch Size	Typical Error (m)	90-th Percentile Error (m)
4×4	0.137	0.212
5×5	0.135	0.208
6×6	0.139	0.213
7×7	0.150	0.230

the image intensity space directly. The proposed method models the true irradiance at each pixel of the patch in the reference image as a random variable, leading to a significant improvement of the estimation accuracy. The formulation of the photometric residual explicitly accounts for the camera response function and lens vignetting (which can be calibrated in advance), as well as unknown illumination gains and biases, which are estimated on a per-feature or per-image basis. Through a detailed experimental evaluation of our algorithm on 50 datasets with high-precision ground truth, we demonstrated that the use of photometric residuals results in increased pose estimation accuracy, with approximately 23% lower estimation errors, on average, in our testing.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (grant no. IIS-1253314 and IIS-1316934), and by Google's project Tango.

REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 260, no. 2, pp. 91–110, Nov. 2004.

[2] E. Rosten, R. Porter, and T. Drummond, "Faster and better: a machine learning approach to corner detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.

[3] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seattle, WA, June 1994, pp. 593–600.

[4] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proceedings of the European Conference on Computer Vision*. Springer, 2014, pp. 834–849.

[5] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, 2014, pp. 15–22.

[6] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," in *arXiv:1607.02565*, July 2016.

[7] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 2007, pp. 3565–3572.

[8] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual-inertial odometry," *International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, May 2013.

[9] H. Yu and A. I. Mourikis, "Vision-aided inertial navigation with line features and a rolling-shutter camera," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, Sept 2015, pp. 892–899.

[10] D. G. Kottas and S. I. Roumeliotis, "Efficient and consistent vision-aided inertial navigation using line observations," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, May 2013, pp. 1540 – 1547.

[11] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proceedings of the IEEE International Conference on Computer Vision*, Washington, DC, USA, 2011, pp. 2320–2327.

[12] V. Usenko, J. Engel, J. Stckler, and D. Cremers, "Direct visual-inertial odometry with stereo cameras," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweeden, May 2016, pp. 1885–1892.

[13] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Sydney, Australia, December 2013.

[14] P. Tanskanen, T. Naegeli, M. Pollefeys, and O. Hilliges, "Semi-direct EKF-based monocular visual-inertial odometry," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, Sept 2015, pp. 6073–6078.

[15] H. Jin, P. Favaro, and S. Soatto, "A semi-direct approach to structure from motion." *The Visual Computer*, vol. 19, no. 6, pp. 377–394, 2003.

[16] F. Devernay and O. Faugeras, "Straight lines have to be straight," *Machine Vision and Applications*, vol. 13, no. 1, pp. 14–24, 2001.

[17] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, 1997, pp. 1106–1113.

[18] D. B. Goldman and J.-H. Chen, "Vignette and exposure calibration and compensation," in *The IEEE International Conference on Computer Vision*, Beijing, China, Oct. 2005, pp. 899–906.

[19] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," in *arXiv:1607.02555*, July 2016.

[20] E. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *International Journal of Robotics Research*, vol. 30, no. 4, pp. 407–430, Apr. 2011.

[21] S. I. Roumeliotis, A. E. Johnson, and J. F. Montgomery, "Augmenting inertial navigation with image-based motion estimation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington D.C, May 2002, pp. 4326–4333.

[22] M. Li and A. I. Mourikis, "Optimization-based estimator design for vision-aided inertial navigation," in *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2012.

[23] J. Stühmer, S. Gumhold, and D. Cremers, "Real-time dense geometry from a handheld camera," in *Proceedings of the 32nd DAGM conference on Pattern recognition*, Berlin, Heidelberg, 2010, pp. 11–20.

[24] J. Kelly and G. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, Jan. 2011.

[25] T. Dong-Si and A. I. Mourikis, "Motion tracking with fixed-lag smoothing: Algorithm and consistency analysis," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, May 2011, pp. 5655 – 5662.

[26] K. Konolige and M. Agrawal, "FrameSLAM: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066 –1077, Oct. 2008.

[27] K. Konolige, M. Agrawal, and J. Sola, "Large-scale visual odometry for rough terrain," in *Proceedings of the International Symposium of Robotics Research*, Flagstaff, AZ, Nov. 2011, pp. 201–212.

[28] A. Delaunoy and M. Pollefeys, "Photometric bundle adjustment for dense multi-view 3D modeling," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 1486–1493.

[29] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct EKF-based approach." in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, 2015, pp. 298–304.

[30] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation," Dept. of Computer Science & Engineering, University of Minnesota, Minneapolis, MN, Tech. Rep. 2005-002, Mar. 2005.

[31] K. Daniilidis, "Hand-eye calibration using dual quaternions," *International Journal of Robotics Research*, vol. 18, no. 3, pp. 286–298, 1999.