Super-Resolution 3D Tracking and Mapping

Maxime Meilland and Andrew I. Comport
I3S CNRS Laboratory
University of Nice Sophia Antipolis
surname@i3s.unice.fr

***Abstract*—This paper proposes a new visual SLAM technique that not only integrates 6 degrees of freedom (DOF) pose and dense structure but also simultaneously integrates the colour information contained in the images over time. This involves developing an inverse model for creating a super-resolution map from many low resolution images. Contrary to classic super-resolution techniques, this is achieved here by taking into account full 3D translation and rotation within a dense localisation and mapping framework. This not only allows to take into account the full range of image deformations but also allows to propose a novel criteria for combining the low resolution images together based on the difference in resolution between different images in 6D space. Another originality of the proposed approach with respect to the current state of the art lies in the minimisation of both colour (RGB) and depth (D) errors, whilst competing approaches only minimise geometry. Several results are given showing that this technique runs in real-time (30Hz) and is able to map large scale environments in high-resolution whilst simultaneously improving the accuracy and robustness of the tracking.**

I. Introduction

The problem of dense real-time localisation and mapping within complex environments is a challenge for a wide range of applications ranging from robotics to augmented reality. In this paper the aim is to be able to interact in real-time with the surfaces of the environment so dense approaches are necessary. This work is undertaken as part of a French DGA Rapid project named Fraudo which requires dense localisation and mapping in real-time using a low cost RGB-D sensor, so as to allow path planning for a mobile robot to traverse uneven ground and surfaces autonomously. The goal is therefore to develop an efficient, accurate and robust *dense visual model* for localisation and mapping. As in all SLAM problems, in order to estimate the unknown maps using a moving sensor, it is necessary to simultaneously estimate the pose of the sensor.

The stated objectives require real-time computational efficiency so several bodies of literature are not considered in this short review but are noted to have overlapping approaches. In particular, the large volume of literature associated with off-line post-production techniques such as Structure From Motion (SFM) and multi-view video [1], [2], [3] have similar problems but perform lengthy calculations using all the data simultaneously. 3D volumetric approaches from the computer graphics literature are also very relevant [4]. Equally, we focus on approaches which look at full 6D transformations including rotation and translation since we consider this to be essential. Even so there are many interesting works which have looked at dense approaches in 2D including optic flow [5] or piecewise dense models such as affine or planar geometry.

In the past ten years a lot of work has been carried out to perform robust real-time 6D localisation and mapping. In particular we can note that the large majority of visual SLAM approaches have used *feature-based techniques* combined with depth and pose estimation [6], [7], [8]. Unfortunately these approaches still are based on an error prone feature extraction step and are not suited to interact with surfaces since they only provide a sparse set of information and do not provide any information about the dense structure of the surface. Amongst the various RGB-D systems, feature based methods include [9], [10], [11]. All of these methods rely on an intermediary estimation processes based on detection thresholds. This feature extraction process is often badly conditioned, noisy and not robust therefore relying on higher level robust estimation techniques. Furthermore, it is necessary to match these features between images over time which is another source of error (feature mapping is not necessarily one-to-one). One popular misconception is that feature-based approaches such as [12] are *direct* appearance-based. Even if these approaches extract features from the intensity information (appearance), they do this via an intermediary estimation process.

More recently, dense techniques have started to become popular and several groups have demonstrated superior real-time performance with commodity hardware. In particular, an early work performing dense 6D SLAM in real-time over large distances [13] was based on minimising an intensity in image key-frames (this approach currently has the best performance on the benchmark of [14]). Other photometric approaches include [15] which looks at fully dense omnidirectional spherical RGB-D sensors. Alternatively, other approaches have focused only on geometry [16], [17], [18]. In the latter truncated signed distance functions (TSDF) are used to define depth integration in a volumetric space and a classic Iterative Closest Point (ICP) is used to estimate the pose. Recent contributions have included using a moving TSDF with ICP [19]. Uniquely geometric approaches are also common to time-of-flight range sensors [20]. Unfortunately the techniques described here either limit themselves to photometric optimisation or geometric optimisation. Neglecting one or the other means that important characteristics are overlooked in terms of robustness, efficiency and precision. It can be noted, however, that in [21], a benchmark test was used to compare both approaches and it was shown that the photometric approach is more precise.

Very few techniques have considered *directly* optimising an error on *both* intensity and depth images. In [22] a direct ICP technique was proposed which minimises colour and depth simultaneously using an image-based approach but super-resolutions were not considered. Alternatively, in [23] both errors were minimised using a volumetric approach based on Octomap but only very small workspaces and resolutions were considered. There are several arguments for and against each approach. In the image based case the resolution of the map is a function of the path taken to acquire it, whereas the volumetric approach is invariant to the path used. In that way the volumetric approach is unable to easily capture the non-linear variation of the image resolution which depends on a particular camera trajectory. Furthermore, incremental volumetric approaches like [18], [19] do not allow to easily perform loop closure correction since the accumulated drift is definitively integrated into the model. More importantly, it should be noted that none of these techniques have tried to "integrate the photometric intensity information", *i.e.* only pose and depth parameters have been estimated.

To investigate models to integrate the image intensity function over time we turn to super-resolution (SR) approaches. In this field a great amount of research has been carried out in the past, however, this has mainly focused on applications such as photography or surveillance so as to obtain better 2D images. More particularly, super-resolution is the art of reconstructing higher resolution images, from a set of lower resolution images. In the most general case, these images are captured from different viewpoints, under different lighting conditions and with sensors of varying resolutions. See Figure 1 for an overview of the image degradation and image reconstruction pipelines. Since the paper of [24], super resolution has been extensively studied in the computer vision community, however, most of the research only considers small relative motion between the input images and the major contributions are focused on how to fuse previously registered images [25], [26], [5]. Furthermore, the registration techniques are mainly 2D and do not take into account knowledge about the dense depth maps of the scene. Several tutorials of these approaches are available which give basic underlying models and principles [27] and more recent approaches aim at extending them such as [28] to perform spatially adaptive block-based super-resolution. Current volumetric approaches [17], [18] are not super-resolution because the integration is performed and linearised in 3D and no colour intensity error is minimised. As such they do not consider increasing the resolution of the images in sensor space which would require practically infeasible volume sizes.

In this paper we propose an approach to not only simultaneously estimate the 6D pose along with the dense 3D map but also the photometric colour in a super-resolution format. This is achieved by considering an inverse compositional approach which is efficient for real-time performance since it allows a maximum of computations to be performed a priori. This differs from the classic super-resolution pipeline as is shown in Figure 1(a). In the model proposed here, direct image-based tracking is used to align the images in 6D while several low resolution images are combined together and integrated to estimate a target high-resolution image (SR). The low resolution (LR) images are combined by weighting their contributions based on a distance to an ideal "virtual image". This virtual image is translated and rotated in such a way that it has the same effective resolution as the high-resolution image even if its resolution is less. In this way images are considered better if they are closer to the same resolution as the target image. Logically, integrating images further from the target resolution will either degrade images due to a loss in resolution or introduce aliasing due to sampling errors.

The remainder of the paper is set out as follows. In Section II an overview is first given for the super-resolution process. In Section III the dense SLAM algorithm is defined. In Section IV-B a simulator is used to obtain a ground truth and evaluate the approach. In Section IV-C real-time images are used to perform super-resolution.

II. Observation Model

Consider a calibrated RGB-D sensor with a colour brightness function $\mathbf{I} : \Omega \times \mathbb{R}^+ \rightarrow \mathbb{R}^+; (\mathbf{p}, t) \mapsto \mathbf{I}(\mathbf{p}, t)$ and a depth function $\mathbf{D} : \Omega \times \mathbb{R}^+ \rightarrow \mathbb{R}^+; (\mathbf{p}, t) \mapsto \mathbf{D}(\mathbf{p}, t)$, where $\Omega = [1, n] \times [1, m] \subset \mathbb{R}^2$ and where $\mathbf{P} = (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{nm}) \in \mathbb{R}^{2\times \mathrm{nm}} \subset \Omega$ are pixel locations within the image acquired at time t and $n \times m$ is the dimension of the sensor's images. It is convenient to consider the set of measurements in vector form such that $\mathbf{I}(\mathbf{P}, t) \in \mathbb{R}^{+\mathrm{nm}}$ and $\mathbf{D}(\mathbf{P}, t) \in \mathbb{R}^{+\mathrm{nm}}$. Note that t and $\mathbf{P}$ may be omitted in the following variables for clarity.

Consider now a RGB-D frame, denoted also an *augmented image* [15], to be the set containing both brightness and depth $\boldsymbol{\mathcal{I}}(t) = \{\mathbf{I}(t), \mathbf{D}(t)\}$. $\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{nm}) \in \mathbb{R}^{3\times \mathrm{nm}}$ is defined as the matrix of 3D vertices related to the surface according to the following point-depth back-projection:

$$\mathbf{v}_i = \tilde{\mathbf{K}}^{-1}\mathbf{p}_i\mathbf{D}(\mathbf{p}_i, t), \tag{1}$$

where $\tilde{\mathbf{K}}$ is the intrinsic matrix of the depth camera.

$\boldsymbol{\mathcal{I}}$ will be called the *current* frame and $\boldsymbol{\mathcal{I}}^*$ the *reference* frame. A superscript $*$ will be used throughout to designate the reference view (or super-resolution frames), and an overscript $\sim$ will be used to differentiate depth from image variables.

Now consider a set of LR augmented images acquired at different times $\{\boldsymbol{\mathcal{I}}(1), \boldsymbol{\mathcal{I}}(2), \ldots, \boldsymbol{\mathcal{I}}(N)\}$, which observe the same scene from different 3D poses. The SR process consists in simultaneously registering and fusing the images onto an augmented SR frame $\boldsymbol{\mathcal{I}}^* \in \mathbb{R}^{2\times \mathrm{q}\times r}$, where $q \times r$ is the resolution of the SR reference images, such that:

$$\begin{aligned}\mathbf{I}^* &= \beta\left(\textstyle\sum_t^N \mathbf{C}(t)\mathbf{I}\left(w\left(\overline{\mathbf{T}}_t, \mathbf{V}^*; \mathbf{K}, \mathbf{S}\right), t\right) + \boldsymbol{\eta}_t, \mathbf{B}^{-1}\right)\\ \mathbf{D}^* &= \beta\left(\textstyle\sum_t^N \tilde{\mathbf{C}}(t)\mathbf{D}\left(w\left(\overline{\mathbf{T}}_t, \mathbf{V}^*; \tilde{\mathbf{K}}, \mathbf{S}\right), t\right) + \tilde{\boldsymbol{\eta}}_t, \tilde{\mathbf{B}}^{-1}\right)\end{aligned} \tag{2}$$

where each matrix $\overline{\mathbf{T}} = (\bar{\mathbf{R}}, \bar{\mathbf{t}}) \in \mathbb{SE}(3)$ is the true pose of a RGB-D camera relative to the reference position (which are not known). Throughout, $\mathbf{R} \in \mathbb{SO}(3)$ is a rotation matrix and $\mathbf{t} \in \mathbb{R}(3)$ the translation vector. The matrix $\mathbf{S} \in \mathbb{R}^{3\times 3}$ is the up-sampling matrix, $\boldsymbol{\mathcal{K}} = \{\mathbf{K}, \tilde{\mathbf{K}}\} \in \mathbb{R}^{2\times 3\times 3}$ are the intrinsic matrices of the colour and depth cameras, $\boldsymbol{\mathcal{C}}(t) =$

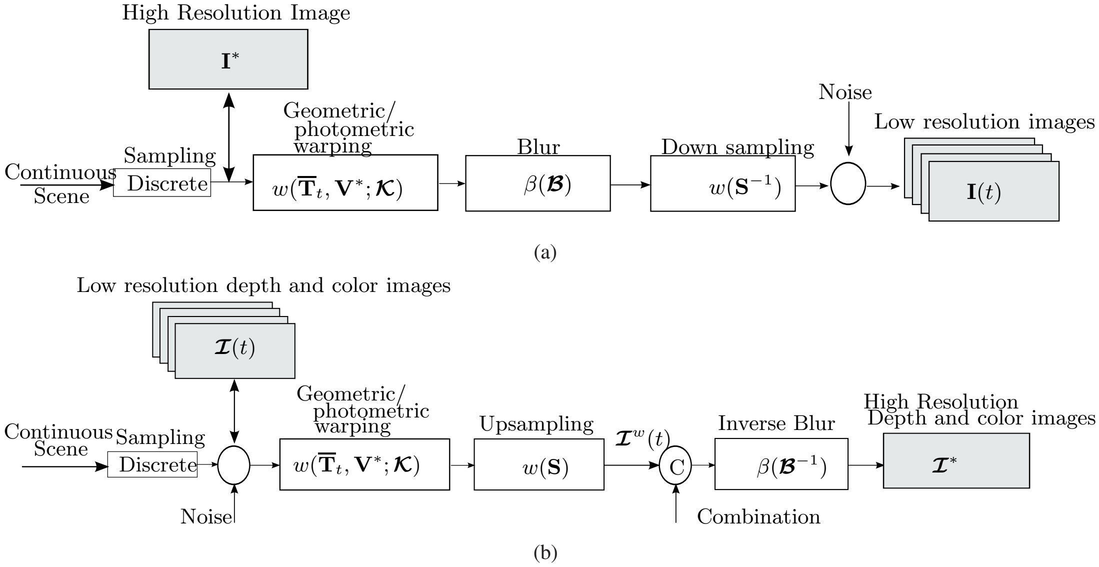

High Resolution Image
$\mathbf{I}^*$
Noise
Low resolution images
Geometric/
photometric
warping
Sampling
Blur
Down sampling
Continuous
Scene
Discrete
$w(\overline{\mathbf{T}}_t, \mathbf{V}^*; \boldsymbol{\mathcal{K}})$
$\beta(\boldsymbol{\mathcal{B}})$
$w(\mathbf{S}^{-1})$
$\mathbf{I}(t)$
(a)
Low resolution depth and color images
$\boldsymbol{\mathcal{I}}(t)$
Geometric/
photometric
warping
Upsampling
$\boldsymbol{\mathcal{I}}^w(t)$
Inverse Blur
High Resolution
Depth and color images
Continuous
Scene
Sampling
Discrete
$w(\overline{\mathbf{T}}_t, \mathbf{V}^*; \boldsymbol{\mathcal{K}})$
$w(\mathbf{S})$
C
$\beta(\boldsymbol{\mathcal{B}}^{-1})$
$\boldsymbol{\mathcal{I}}^*$
Noise
Combination
(b)

Fig. 1. (a) The image degradation pipeline (forward compositional). On the left an imaging sensor samples the incoming light rays to acquire a SR image. This image is at a particular pose in space and the warping transforms the image. Optical, motion and sensor blur then further degenerate the image before it is down-sampled to produce a low resolution image. (b) The image generation pipeline (inverse compositional). Several low resolution images are sampled from a continuous light field. The images are combined via their weighting $\boldsymbol{\mathcal{C}}$ based on their distance to the ideal image with the same resolution. The low resolution images are transformed to a common reference frame. The images are up-sampled and then inverse blurring is applied.

$\{\mathbf{C}(t), \tilde{\mathbf{C}}(t)\} \in \mathbb{R}^{2\times qr\times qr}$ are the diagonal combination matrices of image at time t, η and $\tilde{\eta}$ are Gaussian noise vectors, and $\boldsymbol{\mathcal{B}} = \{\mathbf{B}, \tilde{\mathbf{B}}\}$ are the blur or inverse blur matrices of a given radius. The warping function $w(\cdot)$, the blur function $\beta(\cdot)$ and these variables will now be defined following the processing pipeline of Figure 1(b) from left to right.

1) **Intensity and depth interpolation:** The SR warped image $\mathbf{I}^w$ and depth-map $\mathbf{D}^w$ of dimensions $q\times r$ are obtained by re-sampling the aligned LR images as

$$\boldsymbol{\mathcal{I}}^w(\mathbf{P}^*, t) = \boldsymbol{\mathcal{I}}\left(\mathbf{P}^w, t\right), \tag{3}$$

where $\mathbf{P}^w$ are the projected warped and up-sampled points from equations (4) and (6) given in the following. In practice the depth interpolation functions are optimised and computed differently as in [22] and a bi-linear technique is used. In the Figure 1(b) the transformation blocks are shown separately, however, in practice the transformations are all associative and the images are only interpolated once.

2) **Geometric warping - motion model:** From the next processing block the LR images are transformed from left to right in Figure 1(b), however, the geometric points are warped from right to left so as to interpolate the intensities and depths at the location corresponding to the SR pixels (interpolation is only performed once). The motion model $w(\overline{\mathbf{T}}, \mathbf{v}_i^s; \mathbf{K})$ is therefore a 3D point warping function, which is related to the 3D pose $\overline{\mathbf{T}}$ of the camera and to a scene vertex $\mathbf{v}_i^s$:

$$\mathbf{p}_i^w = \frac{\mathbf{K}(\overline{\mathbf{R}}\mathbf{v}_i^s + \overline{\mathbf{t}})}{\mathbf{e}_3^\top \mathbf{K}(\overline{\mathbf{R}}\mathbf{v}_i^s + \overline{\mathbf{t}})}, \tag{4}$$

where $\mathbf{v}_i^s$ is obtained by applying equation (1) to the sub-sampled pixel $\mathbf{p}_s$ given later in equation (6). $\mathbf{e}_3$ is a unit vector with the third component equal to 1.

3) **Image up-sampling :** The next block involves the up-sampling of the LR image to the SR image. As for the motion model, the high resolution pixels coordinates are transformed from right to left in Figure 1(b). This consists in warping the reference SR pixels by a diagonal homography scaling matrix:

$$\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s \end{bmatrix}, \tag{5}$$

where s is the desired scale factor. A sub-sample pixel in the LR image is then obtained by the following homographic warp that computes sub-pixel coordinates in the LR image:

$$\overline{\mathbf{p}}^s = \frac{\mathbf{S}\overline{\mathbf{p}}^*}{\mathbf{e}_3^\top \mathbf{S}\overline{\mathbf{p}}^*}, \tag{6}$$

where $\overline{\mathbf{p}}^s = ({\mathbf{p}^s}^\top, 1)^\top$ is the homogeneous coordinate of the SR pixel $\mathbf{p}^*$ in the LR space. The relationship between the dimensions of the LR and SR images is subsequently $q = sn$ and $r = sm$. Note we assume that the physical SR sensor size remains the same as the LR sensor, but its resolution is increased (SR and LR pixels units are not the same).

4) **Combination matrices:** The matrices $\mathbf{C}(t)$ and $\tilde{\mathbf{C}}(t)$ are normalised diagonal "combination" matrices, $\sum_t^N \mathbf{C}(t) = \mathbf{I}$, that allow to linearly combine the input depth-map and color images into consistent high resolution ones. This will be shown to penalise a difference in image resolution in the next Section.

5) **Blur:** The function $f(\mathbf{I}^w, \mathbf{B}^{-1})$, is a filter which performs image deconvolution. This will be assumed to be a post-processing step of the reconstructed SR image, that can be achieved using for example a Wiener filter.

A. Image resolution distance function

One of the main contributions of this paper is based on how the LR images are combined to form a SR image. Classic techniques average the aligned images using a smoothing point-spread function [27]. This naive approach has the effect of simply considering the combination matrices to be $\mathbf{C}(\cdot) = \mathbf{I}$. Clearly, this results in a simple average of the input warped images. This often yields a blurred reconstruction since the images seen with a highly different resolution than the SR image are treated the same as those which contain as much detail as those seen by the SR camera. In reality though, the images undergo full 3D transformation and non-linear light field sampling effects are hard to model. To solve this, the aim is to define a distance function with allows to *favour closer effective resolutions* of the LR images w.r.t. the SR image.

The following will show that a LR camera can undergo a 3D transformation with respect to the SR image such that it sees the same effective light rays in space (i.e. the same resolution). This also means that we can compute a set of "optimal virtual images", with the same resolution as the LR image, such that it intersects the same viewing cones as the SR image. Naively, this can be seen as translating the camera toward the scene (rotation also plays a role) so that it sees an effective higher resolution (even if it does not cover the same field of view as the SR camera).

To better understand, consider the Figure 2. The SR image is defined by the frame $\mathbf{T}^*$. The current LR image which must be used to generate a part of the SR image is defined in Frame $\mathbf{T}_t$. Both the LR and SR images observe a vertex $\mathbf{v}^* \in \mathbb{R}^3$ of the scene. The light reflected off the surface at $\mathbf{v}^*$ forms cones in space that are projected onto the SR and LR images respectively. Now consider moving a virtual camera defined by the frame $\mathbf{T}_o$ and with the same resolution as the LR image. This camera can move in 3D via its homogeneous transformation matrix $\mathbf{T}_o = (\mathbf{R}_o, \mathbf{t}_o)$.

The first goal is to determine the 3D pose of the virtual camera such that it has the same effective resolution as the SR image. For each viewing cone, this is equivalent to finding the intersection of the virtual LR image plane with the SR viewing cone such that it sees the same surface area. This area is equivalent when the inverse of the scaling homography $\mathbf{S}^{-1}$, from (5), is equal to the planar homography $\mathbf{H} = \left(\mathbf{R}_o - d^{-1}\mathbf{t_o}\mathbf{n}^\top\right)$. This gives the following constraint on $\mathbf{R}_o$ and $\mathbf{t}_o$:

$$\mathbf{S}^{-1} - \left(\mathbf{R}_o - d^{-1}\mathbf{t_o}\mathbf{n}^\top\right) = \mathbf{0}. \tag{7}$$

The planar homography $\mathbf{H}$ is related to the unknown pose $\mathbf{T}_o$ at which the virtual image intersects the viewing cone and the local surface plane $\boldsymbol{\pi} = (\mathbf{n}^\top, d)^\top$. The viewing cone intersects the 3D surface at vertex $\mathbf{v}^*$ with a certain radius. This forms the plane $\boldsymbol{\pi}$ tangent to the surface with the normal $\mathbf{n}$. This normal is known from the dense 3D map, and is obtained by a local cross product on the image grid. $\mathbf{t}_o$ is the translation vector of the virtual camera and d is the distance between the camera centre of projection and the plane $d = |\mathbf{n}^\top \mathbf{v}^*|$.

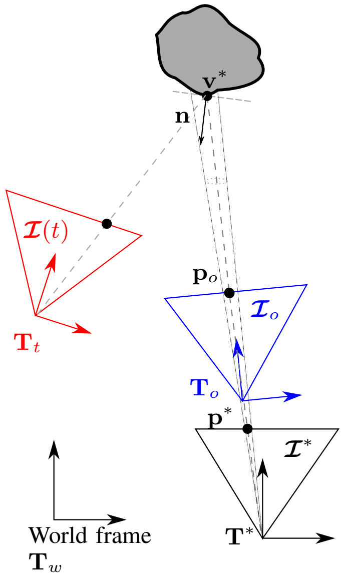


Fig. 2. $\mathbf{T}^*$ is the SR reference image, $\mathbf{T}_o$ is the optimal virtual camera pose (with the same effective resolution of the vertex $\mathbf{v}^*$) and $\mathbf{T}_t$ is a LR frame.

Next we determine $\mathbf{R}_o$ from the intersection of the virtual image with the viewing cone. Since this intersection is invariant to rotations around the Z axis, only the other two rotations need to be computed. In practice, the two remaining rotations are set such that the optical axis the virtual camera is in the direction of the viewing ray of the SR image (for each pixel). This means that the virtual camera is centred on each pixel in the SR image which helps avoid optical lens distortion effects and ensures overlap (see $\mathbf{T}_o$ in Figure 2). Subsequently the full rotation matrix $\mathbf{R}_o = (\mathbf{r}_{ox}^\top, \mathbf{r}_{oy}^\top, \mathbf{r}_{oz}^\top)^\top$ is defined such that

$$\mathbf{r}_{oz} = \mathbf{v}^* \|\mathbf{v}^*\|^{-1}, \quad \mathbf{r}_{ox} = [\mathbf{r}_y^*]_\times \mathbf{r}_{oz}, \quad \mathbf{r}_{oy} = [\mathbf{r}_{oz}]_\times \mathbf{r}_{ox},$$

where $[.]_\times$ represents the skew symmetric matrix operator.

Finally the virtual camera translations are obtained by solving (7) for the translation vector as:

$$\mathbf{t}_o = d(\mathbf{R}_o - \mathbf{S}^{-1})\mathbf{n}.$$

Given $\mathbf{T}_o$ and $\mathbf{T}_t$, it is possible to define a distance metric between each LR pixel and each ideal pixel associated to $\mathbf{v}_i^*$. This distance transforms directly into a weighting coefficient for combining the LR intensities for each image t as

$$\mathbf{C}_{ii}(t) = \left(\|(\mathbf{T}_t - \mathbf{T}_o)\overline{\mathbf{v}}_i^*\| + \epsilon\right)^{-1}, \tag{8}$$

where $\mathbf{C}_{ii}(t)$ is a diagonal element of $\mathbf{C}(t)$. ϵ is a noise constant and $\overline{\mathbf{v}}_i^* = (\mathbf{v}^{*\top}, 1)^\top \in \mathbb{R}^4$ is the homogeneous vertex vector.

It can be seen that this distance constrains 5DOF (i.e. not the rotation around Z). Also if the LR current image moves towards the optimal resolution (in translation and rotation) then the error is zero whilst as it moves away the error increases. The scale factor which combines the rotational and translation components is determined by the vertex $\overline{\mathbf{v}_i^*}$ on the surface.

B. Depth weighting coefficients

A theoretical random error model proposed in [29] is used for weighting depths. The corresponding weighting coefficient of the pixel $\mathbf{p}_i$ is defined by

$$\tilde{\mathbf{C}}_{ii}(t) = \frac{fb}{\sigma_d}\mathbf{D}(\mathbf{p}_i, t)^{-2}, \tag{9}$$

where f is the focal length of the depth camera, b is the baseline and σ_d is the standard deviation of the expected disparity error.

III. Super-resolution visual SLAM

A. Cost function

The SR visual SLAM problem is defined here to be that which estimates, incrementally, the set of camera poses $\mathbf{T}_t(\mathbf{x}_t)$ whilst simultaneously estimating the SR depth image $\mathbf{D}^*$ and the SR intensity measurements $\mathbf{I}^*$ from a set of LR images. This is achieved by considering the following photometric and depth errors:

$$\begin{array}{l}\mathbf{e_I} = \sum_t^N \mathbf{C}(t)\left(\mathbf{I}^* - \mathbf{I}\left(w\left(\widehat{\mathbf{T}}_t\mathbf{T}(\mathbf{x}_t), \mathbf{V}^*, \mathbf{S}\right), t\right)\right) \\ \mathbf{e_D} = \sum_t^N \tilde{\mathbf{C}}(t)\left(\mathbf{D}^* - \mathbf{D}\left(w\left(\widehat{\mathbf{T}}_t\mathbf{T}(\mathbf{x}_t), \mathbf{V}^*, \mathbf{S}\right), t\right)\right)\end{array}, \tag{10}$$

where it is supposed that for each pose there exists an incremental pose that combines homogeneously with the global pose to give the true transformation: $\exists\tilde{\mathbf{x}}_i : \widehat{\mathbf{T}}_t\mathbf{T}(\tilde{\mathbf{x}}_t) = \overline{\mathbf{T}}_t$. The full state vector representing the unknowns is then $[\mathbf{I}^*, \mathbf{D}^*, \mathbf{x}_1, \ldots, \mathbf{x}_N]$. Non-linear optimisation of this error can then be decomposed via marginalisation into two separate minimisation phases which are performed iteratively for each LR input image: i.e. pose estimation followed by depth and intensity estimation. This is the optimal formulation for the joint problem assuming that the initial SR depth and intensities measurements are locally close to the solution [22].

B. 3D image registration and tracking

For each current image, the unknown motion parameters $\mathbf{x} \in \mathbb{R}^6$ are defined as:

$$\mathbf{x} = \int_0^1 (\boldsymbol{\omega}, \boldsymbol{\upsilon})dt \in se(3), \tag{11}$$

which is the integral of a constant velocity twist which produces a pose $\mathbf{T}$. The pose and the twist are related via the exponential map as $\mathbf{T} = e^{[\mathbf{x}]_\wedge}$ with the operator $[.]_\wedge$ as:

$$[\mathbf{x}]_\wedge = \begin{bmatrix} [\boldsymbol{\omega}]_\times & \boldsymbol{\upsilon} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}. \tag{12}$$

Thus the pose cost function is then obtained by simultaneously minimising the errors of equation (10) in an iterative robust least square procedure

$$\mathcal{O}(\mathbf{x}) = \lambda_\mathbf{I}^2 \mathbf{e_I}^\top \mathbf{W_I} \mathbf{e_I} + \lambda_\mathbf{D}^2 \mathbf{e_D}^\top \mathbf{W_D} \mathbf{e_D}, \tag{13}$$

where $\lambda_{(.)}$ are weighting scalar gains and where $\mathbf{W}_{(.)}$ are diagonal weighting matrices obtained by M-estimation [30]. The unknown $\mathbf{x}$ is then iteratively estimated using

$$\begin{array}{l}\mathbf{x} = -(\mathbf{J}^\top\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^\top\mathbf{W}\begin{bmatrix}\mathbf{e_I} & \mathbf{e_D}\end{bmatrix}^\top \\ \widehat{\mathbf{T}} \leftarrow \widehat{\mathbf{T}}\mathbf{T}(\mathbf{x}),\end{array} \tag{14}$$

where $\mathbf{J}$ contains the stacked Jacobian matrices of the errors and $\mathbf{W}$ contains the stacked weighting matrices. More details on such a minimisation can be found in [22].

Minimising both errors provides a lot of advantages since photometric and depth information are complementary. In practice the depth error usually offers a larger domain of convergence and fast minimisation. It also allows to track texture-less areas, but is sensitive to noise and may encounter unconstrained scenes. On the other hand, the photometric term allows to track any textured areas with a better precision [21].

C. Super-resolution

Since the matrices $\mathbf{C}(t)$ and $\tilde{\mathbf{C}}(t)$ are diagonal, minimising equation (10) w.r.t. the photometric parameter $\mathbf{I}^*$ can be done independently as new LR images $\mathbf{I}(t)$ are registered and warped onto the SR frame. The update rule is then:

$$\begin{array}{l}\mathbf{C}^*(t) \leftarrow \mathbf{C}^*(t-1) + \mathbf{C}(t) \\ \mathbf{I}^*(t) \leftarrow \big(\mathbf{C}^*(t-1)\mathbf{I}^*(t-1) + \mathbf{C}(t)\mathbf{I}^w(t)\big)\mathbf{C}^*(t)^{-1}\end{array},$$

where $\mathbf{C}^*(t-1)$ is the global intensity cost at time $t-1$ and $\mathbf{I}^w(t)$ is the warped current image after registration. The same procedure is applied for the depth parameter $\mathbf{D}^*$

$$\begin{array}{l}\tilde{\mathbf{C}}^*(t) \leftarrow \tilde{\mathbf{C}}^*(t-1) + \tilde{\mathbf{C}}(t), \\ \mathbf{D}^*(t) \leftarrow \big(\tilde{\mathbf{C}}^*(t-1)\mathbf{D}^*(t-1) + \tilde{\mathbf{C}}(t)\mathbf{D}^w(t)\big)\tilde{\mathbf{C}}^*(t)^{-1}.\end{array}$$

IV. Experiments

A. Real-time implementation

A real-time implementation of the super resolution tracking and mapping algorithm was implemented on the GPU using OpenCL. The algorithm runs at 30 Hz with low resolution input images of size 640×480 pixels and a $4\times$ super-resolution factor on a Nvidia GTX 670 card. The up-scaling factor of $s = 4$ was chosen for real-time purposes but theoretically it is only limited by the Cramer-Rao lower bound.

The entire tracking and mapping pipeline is performed on the GPU, except insignificant linear algebra computations such as the pose matrix update in (14) which is performed on the CPU. A coarse to fine multi-resolution pose estimation approach is employed as detailed in [13]. Since the RGB-D sensor usually provides noisy depth measurements, a bilateral filter is applied to remove noise whilst preserving discontinuities. The filtered depth-map is only used for pose estimation, whilst the raw depth-map $\mathbf{D}$ is used for depth integration, in order to preserve details in the integration process.

B. Simulated results

The algorithm has been tested on a synthetic sequence of images and depth-maps of dimensions 640×480 with ground truth poses, generated from the Sponza atrium model[1]. The sequence is a 20 meter corridor with textured surfaces and complex geometry. The reference image is taken at the beginning of the sequence and then the camera moves along the corridor (see Figure 3). To simulate realistic data, the input depth-maps are perturbed with a Gaussian noise using the

[1] Sponza atrium model, Dabrovic, M and Meinl, F., 2002

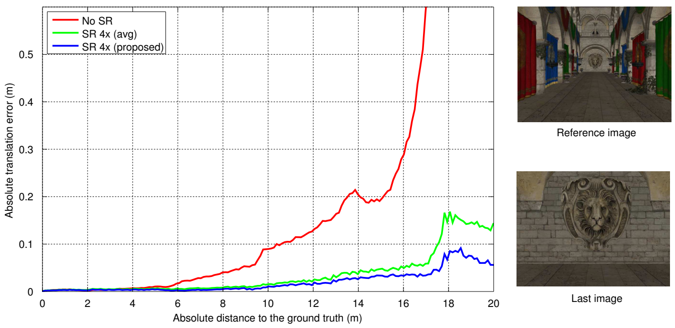


Fig. 3. Absolute translation error with respect to the distance to the reference image. In red, LR tracking without depth and intensity integration. Note that after 10 meters, the LR tracking is not able to track correctly the current image. In green SR tracking with the combination coefficients ($\mathbf{C} = \mathbf{I}$) (equivalent to averaging). In blue the proposed approach. Both SR algorithms reduce the localisation error but the proposed approach clearly outperforms the standard averaging model.

model of equation (9), setting ($\sigma_d = 0.1, f = 530, b = 0.075$). These values where chosen to match the Asus Xtion calibration parameters. The input images are automatically low-passed filtered by the rendering pipeline through mip-mapping.

Three experiments are conducted: First simple tracking is performed without using super-resolution mapping. Then the sequence is tracked with the proposed SR algorithm (scale factor $s = 4$) but only averaging is performed on the intensity ($\mathbf{C}(t) = \mathbf{I}$). Finally SR is performed using the proposed weighting function (8) which takes into account the camera poses and the scene structure.

The plot reported in Figure 3 shows the absolute translation error of the current frame $\mathbf{T}_t$ with respect to the ground truth $\overline{\mathbf{T}}_t$ for each experiment. It can be seen that the SR approach clearly improves the localisation error, by integrating new information as the camera moves along the trajectory. The proposed weighting function also outperforms the standard average, especially when the camera approaches the surface at the end of the corridor. Note that the error peak around $x = 18$ is due the presence of new occlusions (the camera is going under a porch).

C. Experimental results

The visual SLAM algorithm has been successfully tested on a number of real scenes in real-time. The images of Figure 4 report the results of an experiment performed in an office containing a desk, with different objects and books. The RGB-D camera used for this experiment is a calibrated Asus Xtion Pro Live, capturing low resolution images of 640×480 pixels at 30Hz. Super-resolution SLAM is performed in real-time with a scale factor $s = 4$. A first reference image is taken at the beginning of the sequence and the camera is moved around the desk with different motions.

The final super-resolved reference image after optimisation is shown on Figure 4(a). The Figures 4(b) and 4(c) show the Phong shaded surfaces computed from the depth-maps and the surface normals. It can be seen that compared to the original depth-map, the SR one is much more detailed and less noisy.

The second row of Figure 4(a) shows a region of interest of the reference image. Figure 4(d) is extracted from the original LR image, Figure 4(e) is obtained using an averaging of the intensities and Figure 4(f) is obtained using the proposed weighting function. We can visually see that the SR images are highly detailed compared to the original one. The competing averaging approach yields blurry reconstructions since each measurement is averaged independently of the sensor pose, whilst the proposed method produces detailed sharp images.

Thanks to the photometric error, the proposed approach also allows to keep tracking and integrating intensities when the depth camera is totally occluded or too close to the scene (the Asus Xtion minimum range is 30cm) which is not possible with pure ICP techniques such as [18]. Note that the proposed approach is easily applied to any RGB-D sensor including passive stereo, and that larger scenes can be handled as in [13], [15]. Other aspects of this visual SLAM method, such as the ability to track during very rapid motion, or its robustness to camera occlusions are illustrated in the accompanying video[2].

V. Conclusion

In conclusion, this paper has proposed a new super-resolution visual SLAM technique which integrates 6DOF pose and dense structure simultaneously with the colour information contained in the images of a RGB-D sensor over time. A novel inverse model has been provided for creating a super-resolution map from many low resolution images based on a 3D distance criteria which weights the difference in resolution between the low and high resolution images. Additionally this paper shows the importance of minimising both colour and depth errors compared to current ICP approaches which only minimises depth. Experimental results are given showing that this technique runs in real-time (30Hz) and is able to map large scale environments in high-resolution whilst simultaneously improving the accuracy and robustness of the tracking.

Future research in this direction will be focused at using the resolution distance criteria proposed here to better choose the position of the key frames in space. It would also be interesting to use this approach to take into account illumination changes on the surface within a dynamic model.

References

[1] S. Seitz, B. C. amd J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2006.

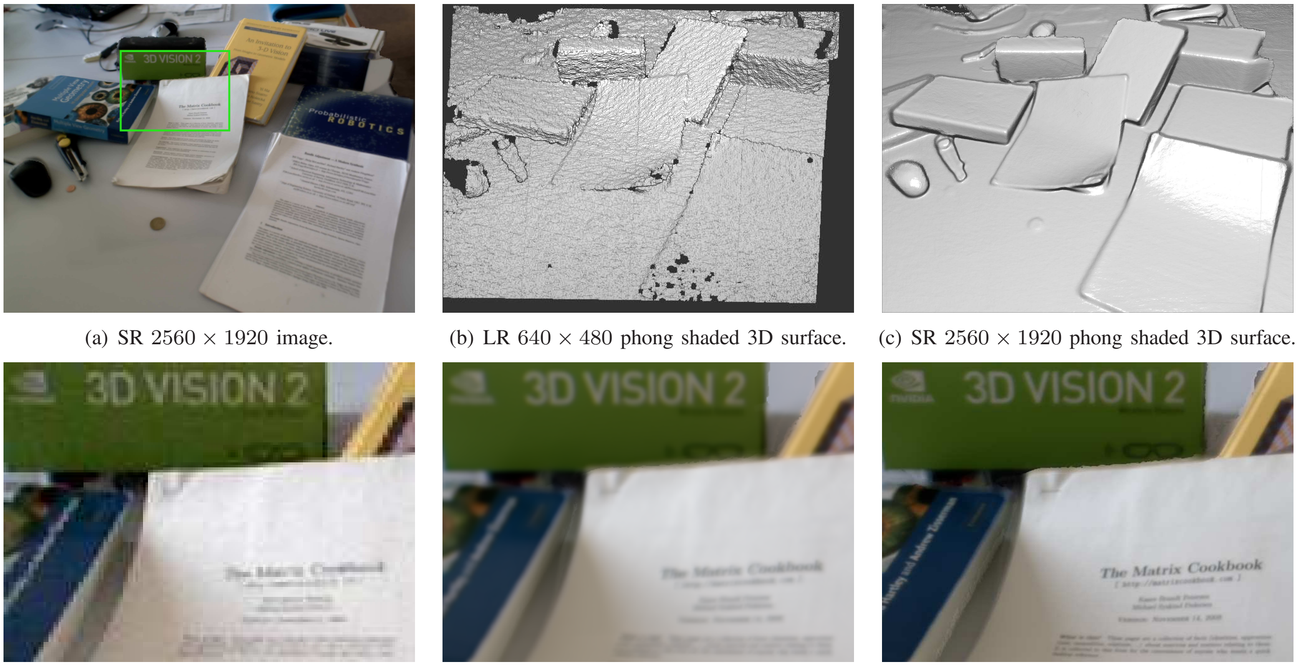
[2] J.-P. Pons and R. Keriven, "Modelling dynamic scenes by registering multi-view image sequences," in *International Conference on Computer Vision and Pattern Recognition*, 2005.

[3] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[4] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Annual Conference on Computer graphics and interactive techniques*, 1996.

[5] S. Baker and T. Kanade, "Super resolution optical flow," Robotics Institute, Tech. Rep., 1999.

[2] http://youtu.be/q51E1NV0Ouw



(a) SR 2560 × 1920 image.

(b) LR 640 × 480 phong shaded 3D surface.

(c) SR 2560 × 1920 phong shaded 3D surface.

(d) Original LR image, ROI.

(e) SR image with average weighting, ROI.

(f) SR image with proposed weighting, ROI.

Fig. 4. (a) Super-resolved image of dimensions 2560 × 1920 pixels. (b) Phong shading of the original 3D surface before optimisation. (c) Phong shading of the SR 3D surface after optimisation. (d) Region of interest of the original image. (e) Region of interest of the SR image with an average weighting after optimisation. (f) Region of interest of the SR image with the proposed weighting after optimisation. It can be seen that the SR SLAM algorithm greatly improves depth measurements as well as intensity measurements. Compared to standard averaging (e) the proposed approach considers the pose of the camera in the weighting function and produces better resolved results.

[6] A. J. Davison and D. W. Murray, "Simultaneous localisation and map-building using active vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 2002.

[7] A. Chiuso, P. Favaro, H. Jin, and S. Soatto, "Structure from motion causally integrated over time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[8] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Real time localization and 3d reconstruction," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2006.

[9] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments," in *International Symposium on Experimental Robotics*, 2010.

[10] N. Engelhard, F. Endres, J. Hess, J. Sturm, and W. Burgard, "Real-time 3d visual slam with a hand-held rgb-d camera," in *RGB-D Workshop on 3D Perception in Robotics*, 2011.

[11] J. Sturm, K. Konolige, C. Stachniss, and W. Burgard, "3d pose estimation, tracking and model learning of articulated objects from dense depth video using projected texture stereo," in *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, 2010.

[12] S. Weik, "Registration of 3-d partial surface models using luminance and depth information," in *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, 1997.

[13] A. I. Comport, E. Malis, and P. Rives, "Accurate quadrifocal tracking for robust 3d visual odometry," in *IEEE International Conference on Robotics and Automation*, 2007.

[14] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition*, 2012.

[15] M. Meilland, A. I. Comport, and P. Rives, "A spherical robot-centered representation for urban navigation," in *IEEE International Conference on Intelligent Robots and Systems*, 2010.

[16] J. Stühmer, S. Gumhold, and D. Cremers, "Real-time dense geometry from a handheld camera," in *DAGM conference on Pattern recognition*, 2010.

[17] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof, "Dense reconstruction on-the-fly," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[18] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *International symposium on mixed and augmented reality*, 2011.

[19] T. Whelan, J. McDonald, M. Kaess, M. Fallon, H. Johannsson, and J. Leonard, "Kintinuous: Spatially extended KinectFusion," in *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, 2012.

[20] S. Schuon, C. Theobalt, J. Davis, and S. Thrun, "Lidarboost: Depth superresolution for tof 3d shape scanning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[21] F. Steinbruecker, J. Sturm, and D. Cremers, "Real-time visual odometry from dense rgb-d images," in *ICCV Workshop on Live Dense Reconstruction with Moving Cameras*, 2011.

[22] T. Tykkala, C. Audras, and A. I. Comport, "Direct iterative closest point for real-time visual odometry," in *ICCV Workshop on Computer Vision in Vehicle Technology*, 2011.

[23] D. Damen, A. Gee, W. Mayol-Cuevas, and A. Calway, "Egocentric real-time workspace monitoring using an rgb-d camera," in *International Conference on Intelligent Robots and Systems*, 2012.

[24] T. S. Huang and R. Y. Tsay, "Multiple frame image restoration and registration," in *Advances in Computer Vision and Image Processing*, 1984.

[25] S. Baker and T. Kanade, "Limits on super-resolution and how to break them," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[26] B. Bascle, A. Blake, and A. Zisserman, "Motion deblurring and super-resolution from an image sequence," in *European Conference on Computer Vision*, 1996.

[27] D. Capel and A. Zisserman, "Computer vision applied to super resolution," *IEEE Signal Processing Magazine*, 2003.

[28] H. Su, L. Tang, Y. Wu, D. Tretter, and J. Zhou, "Spatially adaptive block-based super-resolution," *IEEE Transactions on Image Processing*, 2012.

[29] K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, 2012.

[30] P. Huber, *Robust Statistics*. New york, Wiley, 1981.