Inverse Compositional Estimation of 3D Pose And Lighting in Dynamic Scenes

Yilei Xu, Amit K. Roy-Chowdhury
University of California Riverside, CA 92521

The authors were partially supported by NSF grant IIS-0712253.

Abstract

In this paper, we show how to estimate, accurately and efficiently, the 3D motion of a rigid object and time-varying lighting in a dynamic scene. This is achieved in an inverse compositional tracking framework with a novel warping function that involves a $2D \rightarrow 3D \rightarrow 2D$ transformation. This also allows us to extend traditional two-frame inverse compositional tracking to a sequence of frames, leading to even higher computational savings. We prove the theoretical convergence of this method and show that it leads to significant reduction in computational burden. Experimental analysis on multiple video sequences shows impressive speed-up over existing methods while retaining a high level of accuracy.

Index Terms

inverse composition, tracking, 3D pose, illumination

I. INTRODUCTION

Numerous methods exist for estimating motion and shape of an object from video sequences. Many of them can handle significant changes in the illumination conditions by *compensating* for the variations [1], [2], [3]. However, there do not exist many methods that can *recover* the 3D motion *and* time-varying global illumination conditions from video sequences of moving objects. In this paper, we propose such a method whereby the parameters of an illumination model and the 3D motion are recovered in *continuous time* from video sequences. The work has important applications in a number of areas, most importantly object recognition and inverse rendering.

The goal is achieved by building upon a recently proposed framework for combining the effects of motion, illumination, 3D shape, and camera parameters in a sequence of images obtained by a perspective camera [4]. In one of the most important results on illumination modeling, Basri and Jacobs [5] and Ramamoorthi and Hanrahan [6] independently derived a 9D spherical harmonics based linear representation of the images produced by a Lambertian object with attached shadows. Cast shadow and specular reflectance were not considered. The projection onto this orthonormal basis provided a representation of the global illumination in the image. However, when applied to image sequences, the model requires the knowledge of the surface normals of the object imaged in each frame. Building upon this result, we recently showed that the set of all Lambertian reflectance functions of a *moving* object lies close to a *bilinear* subspace consisting of nine illumination variables and six motion variables [4].

This theory allowed us to develop a mathematical framework for estimating the 3D motion

and illumination parameters from a video sequence with arbitrary lighting changes. A simple method, derived directly from the bilinear space theory for estimating the rigid 3D motion, was presented in [4]. However, this algorithm involved the computation of the bilinear basis in each iteration, which is a huge computational burden. In this paper, we show that it is possible to *efficiently and accurately reconstruct the 3D motion and global lighting parameters from a video sequence within the framework of the inverse compositional algorithm* [7]. This, in turn, provides a quantitative assessment of the accuracy of the theory.

Relation To Previous Work: A well-known approach for 2D motion estimation and registration in monocular sequences is Lucas-Kanade tracking [8]. Building upon this framework, a very efficient tracking algorithm was proposed in [1] by inverting the role of the target image and the template. However, their algorithm can only be applied to restricted class of warps between the target and template (see [7] for details). A forward compositional algorithm was proposed in [9] by estimating an incremental warp for image alignment. Baker et al [7] proposed an inverse compositional (IC) algorithm for efficient implementation of the Lucas-Kanade algorithm to save computational cost in re-evaluation of the derivatives in each iteration. The inverse compositional algorithm was then used for efficiently fitting Active Appearance Models [10] and the well-known 3D morphable model (3DMM) [11] to face images under large pose variations. A dual inverse compositional algorithm was also proposed for dealing with both the geometric and photometric transformations in image registration when lighting varies [12].

None of the above estimate the lighting conditions in the images. An earlier version of 3DMM fitting [13] used a Phong illumination model, estimation of whose parameters in the presence of extended light sources can be difficult. The method in [14] dealt with point sources and did not consider the effect of attached shadows. Specular reflection was taken into consideration in [15], but it dealt with tracking feature points. To handle cast shadows, a physical model incorporating the visible spectrum was introduced for removing the shadows in [16]. Based upon this theory, a shadow resistant image registration method was proposed using the Gauss-Newton method in [17]. Neither of them used an IC approach for motion and lighting estimation.

Our lighting estimation can account for extended lighting sources and attached shadows. Also, we estimate 3D motion, unlike 2D motion in [1], [2], [18], [9], [19]. The warping function in this paper is different from [7], [11] as we explain in Section III. For applications on faces, our approach can be combined with the 3DMM method. Since our inverse compositional approach

estimates 3D motion, it allows us to perform the expensive computations only once every few frames (unlike once for every frame as in the image alignment approaches of [7]). Specifically, these computations are done only when there is a significant change of pose.

Contribution: The following are the major contributions of this paper.

1. We propose a novel 3D model-based warping function for estimating 3D motion and lighting from a video sequence. This involves a $2D \rightarrow 3D \rightarrow 2D$ transformation, which is different from the warping functions used in [7], [11]. This function can be used in future for developing other IC-based tracking algorithms to estimate 3D motion from image sequences.
2. Due to this novel warping function, we are able to extend two-frame IC tracking methods to multiple frames without any significant sacrifice in accuracy.
3. We show that IC approaches can be used not only for estimating 3D motion, but also the time-varying lighting conditions in the scene, including the effects of attached shadows. Existing inverse compositional methods have focused on 2D motion or fitting a 3D model to an image.
4. We rigorously prove the accuracy of the motion and lighting estimates from first principles, analyze the computational savings, and provide results on the numerical correctness of the estimates.

II. Estimating Lighting and Motion in Dynamic Scenes - Direct Approach

In this section, we will briefly review the main results in [4] helping to lay the background and notation for this paper. Let $\mathbf{p} = (\mathbf{T^T}, \mathbf{\Omega^T})^\mathbf{T}, \mathbf{p} \in \mathbb{R}^6$, denote the pose of the object. It was proved that if the motion of the object (defined as the translation of the object centroid $\Delta\mathbf{T} \in \mathbb{R}^3$ and the rotation vector $\Delta\mathbf{\Omega} \in \mathbb{R}^3$ about the centroid in the camera frame) from time t_1 to new time instance $t_2 = t_1 + \delta t$ is small, then upto a first order approximation, the reflectance image $I(x, y)$ at t_2 can be expressed as

$$I_{t_2}(\mathbf{v}) = \sum_{i=1}^{9} l_{i|t_2} b_{i|t_2}(\mathbf{v}), \text{ where } b_{i|t_2}(\mathbf{v}) = b_{i|t_1}(\mathbf{v}) + \mathbf{A}_{t_1}(\mathbf{v}, \mathbf{n})\Delta\mathbf{T} + \mathbf{B}_{t_1}(\mathbf{v}, \mathbf{n})\Delta\mathbf{\Omega}. \quad (1)$$

In the above equations, $\mathbf{v}$ represents the image point projected from the 3D surface with surface normal $\mathbf{n}$, and $b_{i|t_1}(\mathbf{v})$ are the original basis images before motion (precise expression of $b_{i|t_1}(\mathbf{v})$ is defined in [4]). $\mathbf{A}_{t_1}$ and $\mathbf{B}_{t_1}$ contain the structure and camera intrinsic parameters, and are functions of $\mathbf{v}$ and the 3D surface normal $\mathbf{n}$. For each pixel $\mathbf{v}$, both $\mathbf{A}_{t_1}$ and $\mathbf{B}_{t_1}$ are $N_l \times 3$ matrices, where $N_l \approx 9$ for Lambertian objects with attached shadows. Please refer to [4] for the

derivation of (1) and explicit expression for $\mathbf{A}_{t_1}$ and $\mathbf{B}_{t_1}$. For the purposes of this paper, we only need to know the form of the equations. From (1), we see that the new image spans a bilinear space of six motion and approximately nine illumination variables (for Lambertian objects with attached shadows). The basic result is valid for general illumination conditions, but require consideration of higher order spherical harmonics (certain situations will create singularities, which were discussed in [4] but are not important for this paper).

We can express the result in (1) succinctly using tensor notation as

$$\mathcal{I}_{t_2} = \left(\mathcal{B}_{t_1} + \mathcal{C}_{t_1} \times_2 \begin{pmatrix} \Delta\mathbf{T}_{t_2} \\ \Delta\boldsymbol{\Omega}_{t_2} \end{pmatrix} \right) \times_1 \mathbf{l}_{t_2}, \tag{2}$$

where $\times_n$ is called the *mode-n product* [1] [20] and $\mathbf{l}_{t_2} \in \mathbb{R}^{N_l}$, is the vector of $l_{i|t_2}$ components. Thus, the image at t_2 can be represented using the parameters computed at t_1. For each pixel (p, q) in the image, $\mathcal{C}_{klpq|t_1} = [\ \mathbf{A}_{t_1} \quad \mathbf{B}_{t_1}\ ]$ of size $N_l \times 6$. Thus for an image of size $M \times N$, $\mathcal{C}$ is $N_l \times 6 \times M \times N$, $\mathcal{B}_{t_1}$ is a sub-tensor of dimension $N_l \times 1 \times M \times N$, comprising the basis images $b_{i|t_1}(\mathbf{u})$, and $\mathcal{I}_{t_2}$ is a sub-tensor of dimension $1 \times 1 \times M \times N$, representing the image.

Equation (2) provides us an expression relating the reflectance image $\mathcal{I}_{t_2}$ with the illumination coefficients $\mathbf{l}_{t_2}$ and motion variables $\Delta\mathbf{T}, \Delta\boldsymbol{\Omega}$. Letting $\mathbf{m} \triangleq \boldsymbol{\Delta}\mathbf{p} = [\boldsymbol{\Delta}\mathbf{T}^{\mathbf{T}}, \boldsymbol{\Delta}\boldsymbol{\Omega}^{\mathbf{T}}]^{\mathbf{T}}$, we can estimate 3D motion and illumination as

$$(\hat{\mathbf{l}}_{t_2}, \hat{\mathbf{m}}_{t_2}) = \arg \min_{\mathbf{l_{t_2}}, \mathbf{m_{t_2}}} \|\mathcal{I}_{t_2} - (\mathcal{B}_{t_1} + \mathcal{C}_{t_1} \times_2 \mathbf{m}_{t_2}) \times_1 \mathbf{l}_{t_2}\|^2 + \alpha||\mathbf{m}_{t_2}||^2 \tag{3}$$

where $\hat{x}$ denotes an estimate of x. Since the motion between consecutive frames is small, but illumination can change suddenly, we add a regularization term to the above cost function with the form of $\alpha||\mathbf{m}_{t_2}||^2$.

Since the image $\mathcal{I}_{t_2}$ lies approximately in a bilinear space of illumination and motion variables (ignoring the regularization term for now), such a minimization problem can be achieved by alternately estimating the motion and illumination parameters. Assuming that we have tracked the sequence upto some frame at t_1 for which we can estimate the motion (hence, pose) and illumination, we calculate the basis images, $b_{i|t_1}$, at the current pose, and write it in tensor form

[1]The *mode-n product* of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_n \times \ldots \times I_N}$ by a vector $\mathbf{V} \in \mathbb{R}^{1 \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{V}$, is the $I_1 \times I_2 \times \ldots \times 1 \times \ldots \times I_N$ tensor

$$(\mathcal{A} \times_n \mathbf{V})_{i_1 \ldots i_{n-1} 1 i_{n+1} \ldots i_N} = \sum_{i_n} a_{i_1 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} v_{i_n}.$$

$\mathcal{B}_{t_1}$. Similarly, we can compute $\mathcal{C}_{t_1}$ at this pose. Unfolding[2] [20] $\mathcal{B}_{t_1}$ and the image $\mathcal{I}_{t_2}$ along the first dimension, which is the illumination dimension, the illumination can be estimated as

$$\hat{\mathbf{l}}_{t_2} = (\mathcal{B}_{t_1(1)}\mathcal{B}_{t_1(1)}^T)^{-1}\mathcal{B}_{t_1(1)}\mathcal{I}_{t_2(1)}^T. \tag{4}$$

Keeping the illumination coefficients fixed, the bilinear space in equation (2) becomes a linear subspace, i.e.,

$$\mathcal{I}_{t_2} = \mathcal{B}_{t_1} \times_1 \mathbf{l}_{t_2} + \mathcal{G} \times_2 \mathbf{m}_{t_2}, \text{ where } \mathcal{G} = \mathcal{C}_{t_1} \times_1 \mathbf{l}_{t_2}, \tag{5}$$

and motion can be estimated as

$$\hat{\mathbf{m}}_{t_2} = \left(\mathcal{G}_{(2)}\mathcal{G}_{(2)}^T + \alpha\mathbf{I}\right)^{-1}\mathcal{G}_{(2)}(\mathcal{I}_{t_2} - \mathcal{B}_{t_1} \times_1 \mathbf{l}_{t_2})_{(2)}^T, \tag{6}$$

where $\mathbf{I}$ is an identity matrix of dimension 6×6. When we apply the Levenberg-Marquardt method [21] to minimize the difference between the input frame and the rendered frame in (2), we will have exactly the same expression as in (6) with α being the corresponding damping factor. When the regularization term is ignored, the result becomes that of the Gauss-Newton method.

III. Inverse Compositional Estimation of 3D Motion and Illumination

The method described in Section II requires iteration between equations (4) and (6). In each iteration, as pose is updated, the tensors $\mathcal{B}_t$ and $\mathcal{G}_t$ need to be recomputed, which is very expensive computationally (since they require finding the point of intersection of the ray through each point with the 3D surface). In this section, we will derive an inverse compositional approach for efficient and accurate estimation of 3D motion and illumination. We start by showing that (3) is equivalent to a Lucas-Kanade algorithm for estimation of 3D motion and lighting which leads to the inverse compositional approach. Finally, we show how to extend it to a sequence of frames. In keeping the standard notation used in tracking, we assume $\delta t = 1$, and consider two frames at t and $t-1$.

A. Lucas-Kanade Estimation of 3D Motion and Lighting

Let us initially start with the condition that illumination does not change between two frames. We will then consider the varying illumination condition. Also, we ignore the regularization term in (3), which can be easily added back later. The image synthesis process can be considered as

[2]Assume an Nth-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$. The matrix unfolding $\mathbf{A}_{(n)} \in \mathbb{R}^{I_n \times (I_{n+1}I_{n+2}\ldots I_N I_1 I_2 \ldots I_{n-1})}$ contains the element $a_{i_1 i_2 \ldots i_N}$ at the position with row number i_n and column number equal to $(i_{n+1}-1)I_{n+2}I_{n+3}\ldots I_N I_1 I_2 \ldots I_{n-1} + (i_{n+2}-1)I_{n+3}I_{n+4}\ldots I_N I_1 I_2 \ldots I_{n-1} + \cdots + (i_N - 1)I_1 I_2 \ldots I_{n-1} + (i_1 - 1)I_2 I_3 \ldots I_{n-1} + \cdots + i_{n-1}$.

a rendering function of the object at pose $\mathbf{p}$ in the camera frame to the pixel coordinates $\mathbf{v}$ in the image plane as $f(\mathbf{v}, \mathbf{p}_t)$. Using the bilinear model described above, it can be implemented with (5). Given an input image $I_t(\mathbf{v})$, we want to align the synthesized image with it to obtain

$$\hat{\mathbf{p}}_t = \arg\min_{\mathbf{p}_t} \sum_{\mathbf{v}} \left(f(\mathbf{v}, \mathbf{p}_t) - I_t(\mathbf{v})\right)^2, \tag{7}$$

where $\hat{\mathbf{p}}_t$ denotes the estimated pose for this input image $I_t(\mathbf{v})$. This is the cost function of Lucas-Kanade tracking in [7] modified for 3D motion estimation.

Let us now consider the problem of estimating the pose change, $\triangle\mathbf{p}_t = \mathbf{m}_t$, between two consecutive frames, $I_t(\mathbf{v})$ and $I_{t-1}(\mathbf{v})$ as

$$\hat{\mathbf{m}}_t = \arg\min_{\mathbf{m}_t} \sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \mathbf{m}_t) - I_t(\mathbf{v})\right)^2, \text{ and } \hat{\mathbf{p}}_t = \hat{\mathbf{p}}_{t-1} + \hat{\mathbf{m}}_t. \tag{8}$$

The optimization of the above equation can be achieved by assuming a current estimate of $\hat{\mathbf{m}}_t$ is known and iteratively solve for increments $\triangle\mathbf{m}$ ($\triangle\mathbf{m}$ are the increments between two iterations, where multiple iterations will be needed to get $\mathbf{m}_t$) such that

$$\sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \mathbf{m}_t + \triangle\mathbf{m}) - I_t(\mathbf{v})\right)^2 \tag{9}$$

is minimized. Applying the first order Taylor expansion on (9), we can rewrite it as

$$\sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \mathbf{m}_t) + \frac{\partial f(\mathbf{v}, \mathbf{p})}{\partial \mathbf{p}}^{\mathbf{T}} |_{\mathbf{p}=\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t} \triangle\mathbf{m} - I_t(\mathbf{v}) \right)^2. \tag{10}$$

Recall that equation (5) linearizes the image intensity I with respect to the motion parameter $\mathbf{m}$ when illumination parameter $\mathbf{l}$ is fixed. Thus, from equation (5), we have

$$\frac{\partial f(\mathbf{v}, \mathbf{p}(\mathbf{m_t}))}{\mathbf{m}_t}|_{\mathbf{p}(\mathbf{m_t})=\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t} = \frac{\partial f(\mathbf{v}, \mathbf{p})}{\partial \mathbf{p}} \frac{\partial \mathbf{p}(\mathbf{m_t})}{\partial \mathbf{m}_t}|_{\mathbf{p}(\mathbf{m_t})=\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t} = \frac{\partial f(\mathbf{v}, \mathbf{p})}{\partial \mathbf{p}}|_{\mathbf{p}=\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t} = \mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}}, \tag{11}$$

where $\mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}}$ denotes the components of $\mathcal{G}$ at the pixel coordinate $\mathbf{v}$ computed at the pose $\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t$, and $\mathbf{p}(\mathbf{m_t})$ is used to clearly show that pose $\mathbf{p}$ depends on the $\mathbf{m}_t$ (see (8)). Physically, $\mathcal{G}_{\mathbf{v}}$ contains the information of the object structure and the camera model. Since $\mathcal{C}$ is a tensor of size $N_l \times 6 \times M \times N$ and $\mathcal{G} = \mathcal{C} \times_1 \mathbf{l}$, therefore $\mathcal{G}$ is of size $1 \times 6 \times M \times N$. At a specific pixel $\mathbf{v}$, $\mathcal{G}_{\mathbf{v}}$ degenerates to a 6×1 vector. Substituting (11) into (10), taking the derivative with respect to $\triangle\mathbf{m}$ and setting it to be zero, we get

$$\sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \mathbf{m}_t) + \mathcal{G}^{\mathbf{T}}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \triangle\mathbf{m} - I_t(\mathbf{v})\right) \mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} = 0. \tag{12}$$

Then solving for $\triangle\mathbf{m}$, we have

$$\triangle\mathbf{m} = \mathbf{H} \sum_{\mathbf{v}} \mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \left(I_t(\mathbf{v}) - f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \mathbf{m}_t)\right), \text{where } \mathbf{H} = \left[\sum_{\mathbf{v}} (\mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \mathcal{G}^{\mathbf{T}}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}})\right]^{-1}. \tag{13}$$

Let us now reintroduce the illumination variation which was ignored for simplicity of explanation. The image synthesis function f can be replaced with the analytical expression in (2). Although $\mathcal{G}_{\bullet|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}}$ varies with the illumination condition $\mathbf{l}_t$ according to (5), $\mathcal{C}_{\bullet|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}}$ is not a function of $\mathbf{l}_t$. Thus, given $\mathbf{l}_t$, (13) becomes:

$$\begin{aligned} \triangle\mathbf{m} &= \mathbf{H}\sum_{\mathbf{v}}\left(\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \times_1 \mathbf{l}_t\right)\left(I_t(\mathbf{v}) - \mathcal{B}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \times_1 \mathbf{l}_t\right), \\ \text{where} \qquad \mathbf{H} &= \left[\sum_{\mathbf{v}}(\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \times_1 \mathbf{l}_t)(\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}} \times_1 \mathbf{l}_t)^{\mathbf{T}}\right]^{-1}, \end{aligned} \tag{14}$$

which is effectively equation (6) when α is zero. Once motion is known, lighting can be easily estimated by computing $\mathcal{B}$ in (4). Thus, the direct method we described in Section II is equivalent to Lucas-Kanade 3D tracking and illumination estimation algorithm.

B. Inverse Compositional Estimation of 3D Motion and Lighting

In the above method, the motion $\mathbf{m}$ is updated in each iteration and $\mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}}$ needs to be reevaluated. This requires exhaustively visiting every intersection point of each ray with the surface and computing the derivatives, which extracts a huge computational cost. Thus, it is inefficient to use $\mathcal{G}_{\bullet|\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t}$ in each step of motion and lighting estimation.

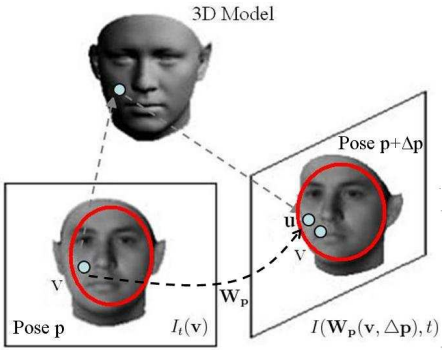


Fig. 1. Illustration of the warping function $\mathbf{W}$. A point $\mathbf{v}$ in image plane is projected onto the surface of the 3D object model. After the pose transformation with $\triangle\mathbf{p}$, the point on the surface is back projected onto the image plane at a new point $\mathbf{u}$. The warping function maps from $\mathbf{v} \in \mathbb{R}^{\mathbf{2}}$ to $\mathbf{u} \in \mathbb{R}^{\mathbf{2}}$. The red ellipses show the common part in both frames that the warping function $\mathbf{W}$ is defined upon.

Let us now introduce a warp operator $\mathbf{W} : \mathbb{R}^{\mathbf{2}} \rightarrow \mathbb{R}^{\mathbf{2}}$ such that, if we denote the pose of $I_t(\mathbf{v})$ as $\mathbf{p}$, the pose of $I_t(\mathbf{W_p}(\mathbf{v}, \triangle\mathbf{p}))$ is $\mathbf{p} + \triangle\mathbf{p}$. Specifically, a 2D point on the image plane is projected onto the 3D object surface. Then we transform the pose of the object surface by $\triangle\mathbf{p}$ and back project the point from the 3D surface onto the image plane. Thus, $\mathbf{W}$ represents the displacement in the image plane due to a pose transformation of the 3D model. Note that this warping involves a 3D pose transformation (unlike [7]). In [11], the warping was from a point on the 3D surface to the image plane, and was used for fitting a 3D model to an image.

We propose a new warping function for the inverse compositional estimation of 3D rigid motion and illumination in video sequence, which is not addressed in [7] or [11].

Using this warp operator, for any frame $I_t(\mathbf{v})$, the cost function (8) can be written as

$$\hat{\mathbf{m}}_t = \arg\min_{\mathbf{m}_t} \sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1}) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t)) \right)^2. \tag{15}$$

Rewriting the cost function (15) in the inverse compositional framework [7], we consider minimizing

$$\arg\min_{\triangle\mathbf{m}} \sum_{\mathbf{v}} \left(f(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, \triangle\mathbf{m}), \hat{\mathbf{p}}_{t-1}) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t)) \right)^2 \tag{16}$$

with the update rule

$$\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t) \leftarrow \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t) \circ \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, \triangle\mathbf{m})^{-1}.^3 \tag{17}$$

We will first derive the solution to (16), then we will prove its equivalence to (15) in Sec. III-C. The compositional operator $\circ$ in (17) means the second warp is composed into the first warp, i.e., $\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m_t}) \equiv \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, \triangle\mathbf{m})^{-1}, -\mathbf{m_t})$.

According to the definition of $\mathbf{W}$, we can approximate $f(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, \triangle\mathbf{m}), \hat{\mathbf{p}}_{t-1})$ in (16) with $f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \triangle\mathbf{m})$[4]. Applying the first order Taylor expansion on it, we have

$$\sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1}) + \frac{\partial f(\mathbf{v}, \mathbf{p})}{\partial \mathbf{p}}|_{\mathbf{p}=\hat{\mathbf{p}}_{t-1}} \triangle\mathbf{m} - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t)) \right)^2. \tag{18}$$

Taking the derivative of (18) with respect to $\triangle\mathbf{m}$ and setting it to be zero, we have

$$\sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1}) + \mathcal{G}^{\mathbf{T}}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \triangle\mathbf{m} - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t)) \right) \mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} = 0. \tag{19}$$

[3]The inverse of the warp $\mathbf{W}$ is defined to be the $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ mapping such that if we denote the pose of $I_t(\mathbf{v})$ as $\mathbf{p}$, the pose of $I_t(\mathbf{W}_{\mathbf{p}}(\mathbf{W}_{\mathbf{p}}(\mathbf{v}, \triangle\mathbf{p}), \triangle\mathbf{p})^{-1})$ is $\mathbf{p}$ itself. As the warp $\mathbf{W}_{\mathbf{p}}(\mathbf{v}, \triangle\mathbf{p})$ transforms the pose from $\mathbf{p}$ to $\mathbf{p} + \triangle\mathbf{p}$, the inverse $\mathbf{W}_{\mathbf{p}}(\mathbf{v}, \triangle\mathbf{p})^{-1}$ should transform the pose from $\mathbf{p} + \triangle\mathbf{p}$ to $\mathbf{p}$, i.e. $\mathbf{W}_{\mathbf{p}}(\mathbf{v}, \triangle\mathbf{p})^{-1} = \mathbf{W}_{\mathbf{p}+\triangle\mathbf{p}}(\mathbf{v}, -\triangle\mathbf{p})$. Thus $\{\mathbf{W}_{\mathbf{p}}\}$ is a group.

[4]This is because $f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \triangle\mathbf{m})$ is the image synthesized at $\hat{\mathbf{p}}_{t-1} + \triangle\mathbf{m}$, while $f(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, \triangle\mathbf{m}), \hat{\mathbf{p}}_{t-1})$ is the image synthesized at $\hat{\mathbf{p}}_{t-1}$ followed with the warp of the pose increments $\triangle\mathbf{m}$. Although illumination is rotated by $\Delta\mathbf{m}$ in $f(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, \triangle\mathbf{m}), \hat{\mathbf{p}}_{t-1})$, for Lambertian objects it is not difficult to show that $f(\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, \triangle\mathbf{m}), \hat{\mathbf{p}}_{t-1}) - f(\mathbf{v}, \hat{\mathbf{p}}_{t-1} + \triangle\mathbf{m}) \sim O(\Delta\mathbf{m}) = o(\Delta\hat{\mathbf{p}}_{t-1})$. Neglecting this amounts to neglecting second order pose variations, which is the same approximation as the one used for the proof of the IC algorithm in Sec. III-C. Thus this substitution is valid for our case.

Solving for $\triangle\mathbf{m}$, we get:

$$\triangle\mathbf{m} = \mathbf{H_{IC}} \sum_{\mathbf{v}} \mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \left(I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t)) - f(\mathbf{v}, \hat{\mathbf{p}}_{t-1}) \right), \text{where } \mathbf{H_{IC}} = \left[\sum_{\mathbf{v}} \mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \mathcal{G}^{\mathbf{T}}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \right]^{-1}. \quad (20)$$

Comparing with equation (13), the derivative $\mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}}$ and Hessian $\mathbf{H_{IC}}$ in (20) do not depend upon the updating variable $\mathbf{m}_t$, which is moved into the warp operator $\mathbf{W}$. The computational complexity of $\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, -\mathbf{m_t})$ will be significantly lower than that of recomputing $\mathcal{G}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}+\mathbf{m_t}}$ and Hessian $\mathbf{H}$ in every iteration (see Section III-F for details on the computational cost).

Reintroducing the illumination variation and following the same derivation as (14), we have

$$\begin{aligned} \triangle\mathbf{m} &= \mathbf{H_{IC}} \sum_{\mathbf{v}} (\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \times_1 \mathbf{l}_t) \left(I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m}_t)) - \mathcal{B}_{\mathbf{v}|\hat{\mathbf{p}}_{t-1}} \times_1 \mathbf{l}_t \right), \\ \text{where} \qquad \mathbf{H_{IC}} &= \left[\sum_{\mathbf{v}} (\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \times_1 \mathbf{l}_t)(\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{\mathbf{t-1}}} \times_1 \mathbf{l}_t)^{\mathbf{T}} \right]^{-1}. \end{aligned} \quad (21)$$

C. Proof of the IC Estimation Algorithm

Using the above update rule, we will now show the equivalence of (16) to (15), which is equivalent to the cost function (8) in the Lucas-Kanade 3D tracking method.

Considering (16), the continuous version of which can be written as

$$\int_V (f(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(v, \triangle\mathbf{m}), \hat{\mathbf{p}}_{t-1}) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(v, -\mathbf{m}_t)))^2 dv, \quad (22)$$

where V is the collection of all the pixels within the image at the pose $\hat{\mathbf{p}}_{t-1}$. Let $u \triangleq \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(v, \triangle\mathbf{m})$, thus $v = \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(u, \triangle\mathbf{m})^{-1} = \mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, -\triangle\mathbf{m})$. Plugging it into (22), we have

$$\int_U (f(u, \hat{\mathbf{p}}_{t-1}) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, -\triangle\mathbf{m}), -\mathbf{m}_t)))^2 \left| \frac{d\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, -\triangle\mathbf{m})}{du} \right| du. \quad (23)$$

Note that with $\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, 0) = u$, it follows that

$$\frac{d\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, -\triangle\mathbf{m})}{du} = 1 + O(\triangle\mathbf{m}) = 1 + o(\mathbf{m}_t) = 1 + o(\triangle\hat{\mathbf{p}}_{t-1}). \quad (24)$$

Recall that $u = \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(v, \triangle\mathbf{m})$, i.e., U is the image of V after warping with $\mathbf{W}$. Since V is the collection of all the pixels within the image at pose $\hat{\mathbf{p}}_{t-1}$, U is the collection of all the pixels within the image at pose $\hat{\mathbf{p}}_{t-1} + \triangle\mathbf{m}$. For a video sequence, the motion $\mathbf{m}$ between the consecutive frames is usually small, thus the increments $\triangle\mathbf{m}$ should be even smaller. With such small increments, the change of the image region should be small, i.e.

$$U = V + O(\triangle\mathbf{m}) = V + o(\mathbf{m}_t) = V + o(\triangle\hat{\mathbf{p}}_{t-1}). \quad (25)$$

Thus U and V differ only in the second order pose variation terms. Also, in the warp composition $\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(\mathbf{u}, -\triangle\mathbf{m}), -\mathbf{m_t})$, the inner warp transforms the pose of the object from $\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}$ to $\hat{\mathbf{p}}_{t-1}$, while the outer warp transforms the pose from $\hat{\mathbf{p}}_{t-1}$ to $\hat{\mathbf{p}}_{t-1}-\mathbf{m}_t$. Thus, it can be simplified as $\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, -\mathbf{m_t}-\triangle\mathbf{m})$. Neglecting the second order variation of the pose with respect to $\hat{\mathbf{p}}_{t-1}$, i.e., neglecting $\triangle\mathbf{m}$ w.r.t. $\hat{\mathbf{p}}_{t-1}$, but not w.r.t. $\mathbf{m}$, we get (see footnote[5] for details)

$$\begin{aligned}\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u, -\mathbf{m}_t-\triangle\mathbf{m})) &\approx \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(u, -\mathbf{m}_t-\triangle\mathbf{m})) + o(\mathbf{m}_t)\\ &= \mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(u, -\mathbf{m}_t-\triangle\mathbf{m})) + o(\triangle\hat{\mathbf{p}}_{t-1}). \qquad (26)\end{aligned}$$

Consequently, using (24), (25) and (26), and neglecting the second order pose variations, (23) can be approximated with

$$\int_V (f(v, \hat{\mathbf{p}}_{t-1}) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(v, -\mathbf{m}_t-\triangle\mathbf{m})))^2 dv. \qquad (27)$$

Note that this assumption of ignoring the second order pose variations is similar to the assumption in [7] of neglecting the second order variation in the parameter set.

Rewriting (27) in the discrete format, we have

$$\arg\min_{\triangle\mathbf{m}} \sum_{\mathbf{v}} \left(f(\mathbf{v}, \hat{\mathbf{p}}_{t-1}) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, -\mathbf{m_t}-\triangle\mathbf{m}))\right)^2, \qquad (28)$$

which is the solution strategy for minimizing (15) using the additive update rule $\mathbf{m}_t \leftarrow \mathbf{m}_t+\triangle\mathbf{m}$. Thus the cost functions (15) and (16) are equivalent, and the inverse compositional update rule can be approximated with the additive rule.

D. Inverse Compositional Estimation Over A Sequence of Frames

The computational complexity in the above derivation is reduced by pre-computing the derivative $\mathcal{G}$ and Hessian $\mathbf{H_{IC}}$ for reuse in each iteration. For the new input frame at time t, although

[5]Consider the warp $\mathbf{W}_{\mathbf{p}+\xi_1}(u, \xi_2)$, where ξ_1 and ξ_2 are both small w.r.t. $\mathbf{p}$. Let $\xi_1 = \varpi_1\triangle\theta_1$ and $\xi_2 = \varpi_2\triangle\theta_2$, where ϖ_1 and ϖ_2 are unit vectors. $\mathbf{x}$ is the 3D coordinate of a vertex on the 3D model. Using an orthographic or weak perspective camera model, the first dimension of the warp can be expressed as $(e^{\xi_1}e^{\mathbf{P}}\mathbf{x})^{(1)} - (e^{\xi_2}e^{\xi_1}e^{\mathbf{P}}\mathbf{x})^{(1)} \approx ((I+\tilde{\varpi_1}\sin\theta_1)e^{\mathbf{P}}\mathbf{x})^{(1)} - ((I+\tilde{\varpi_2}\sin\theta_2)(I+\tilde{\varpi_1}\sin\theta_1)e^{\mathbf{P}}\mathbf{x})^{(1)} = -(\tilde{\varpi_2}\sin\theta_2 e^{\mathbf{P}}\mathbf{x})^{(1)} + o(\xi_1, \xi_2)$, where $\tilde{\varpi}$ denotes the skew symmetric matrix with entries $\begin{pmatrix} 0 & -\varpi^{(3)} & \varpi^{(2)} \\ \varpi^{(3)} & 0 & -\varpi^{(1)} \\ -\varpi^{(2)} & \varpi^{(1)} & 0 \end{pmatrix}$, and the superscript $^{(1)}$ indicates the first dimension of the vector. Similar operations can be applied on the second dimension of warp. Thus, when both ξ_1 and ξ_2 are small terms w.r.t. $\mathbf{p}$, $\mathbf{W}_{\mathbf{p}+\xi_1}(\mathbf{u}, \xi_2) \approx \mathbf{W}_{\mathbf{p}}(\mathbf{u}, \xi_2)$.

$\mathcal{G}_{\bullet|\hat{\mathbf{p}}_t}$ would be close to $\mathcal{G}_{\bullet|\hat{\mathbf{p}}_{t-1}}$, it still needs to be recomputed. To further save computation complexity in the video sequence context, we can apply a similar idea by choosing a cardinal pose $\mathbf{p_c}$, pre-compute the derivatives $\mathcal{G}_{\mathbf{v}|\mathbf{p_c}}$ and $\mathbf{H}_{\mathbf{IC}|\mathbf{p_c}}$, and then reuse them for consequent frames.

Let us consider a sequence of frames $I(\bullet,1),\ldots I(\bullet,t),\ldots I(\bullet,N)$. Without loss of generality, let us assume that the cardinal pose, $\mathbf{p_c}$, is at frame $I(\bullet,1)$, i.e. $\mathbf{p_c}=\hat{\mathbf{p}}_\mathbf{1}$. Assume we already know the estimated motion upto time instance $t-1$, $\hat{\mathbf{m}}_\mathbf{1},\ldots\hat{\mathbf{m}}_{\mathbf{t-1}}$. For the input frame $I_t(\mathbf{v})$, we use the pose transformation operator $\mathbf{W}$ to normalize the pose to the cardinal pose based on $\hat{\mathbf{p}}_\mathbf{1},\ldots\hat{\mathbf{p}}_{\mathbf{t-1}}$, i.e.,

$$\hat{\mathbf{m}}_t=\arg\min_{\mathbf{m}_t}\sum_{\mathbf{v}}\big(f(\mathbf{v},\mathbf{p_c})-I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v},(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t))\big)^2. \tag{29}$$

Rewriting the cost function (29) in the inverse compositional framework, we consider minimizing

$$\arg\min_{\triangle\mathbf{m}}\sum_{\mathbf{v}}\big(f(\mathbf{W}_{\mathbf{p_c}}(\mathbf{v},\triangle\mathbf{m}),\mathbf{p_c})-I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v},(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t))\big)^2. \tag{30}$$

with the update rule

$$\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v},(\mathbf{p_c}-\hat{\mathbf{p}}_{\mathbf{t-1}})-\mathbf{m_t})\leftarrow\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v},(\mathbf{p_c}-\hat{\mathbf{p}}_{\mathbf{t-1}})-\mathbf{m_t})\circ\mathbf{W}_{\mathbf{p_c}}(\mathbf{v},\triangle\mathbf{m})^{-1}. \tag{31}$$

Note that (30) is similar to (16), except that the warping for f is computed at the cardinal pose. Following the derivation of equations (18) - (21) and reintroducing the illumination variation, we have

$$\begin{aligned}\triangle\mathbf{m} &= \mathbf{H_{IC}}\sum_{\mathbf{v}}(\mathcal{C}_{\mathbf{v}|\mathbf{p_c}}\times_1\mathbf{l}_t)\big(I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v},(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t))-\mathcal{B}_{\mathbf{v}|\mathbf{p_c}}\times_1\mathbf{l}_t\big),\\ \text{where}\quad \mathbf{H_{IC}} &= \left[\sum_{\mathbf{v}}(\mathcal{C}_{\mathbf{v}|\mathbf{p_c}}\times_1\mathbf{l}_t)(\mathcal{C}_{\mathbf{v}|\mathbf{p_c}}\times_1\mathbf{l}_t)^{\mathbf{T}}\right]^{-1}.\end{aligned} \tag{32}$$

The proof of (32) can be done in a way similar to that of Section III-C. Rewriting (30) in continuous domain and substituting $u\triangleq\mathbf{W}_{\mathbf{p_c}}(v,\triangle\mathbf{m})$ (conversely, $v=\mathbf{W}_{\mathbf{p_c}}(u,\triangle\mathbf{m})^{-1}=\mathbf{W}_{\mathbf{p_c}+\triangle\mathbf{m}}(u,-\triangle\mathbf{m})$),

$$\int_U(f(u,\mathbf{p}_c)-I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\mathbf{p_c}+\triangle\mathbf{m}}(u,-\triangle\mathbf{m}),(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t)))^2|\frac{d\mathbf{W}_{\mathbf{p_c}+\triangle\mathbf{m}}(u,-\triangle\mathbf{m})}{du}|du. \tag{33}$$

Assuming that pose $\hat{\mathbf{p}}_{t-1}$ does not deviate from $\mathbf{p_c}$ too much, and from footnote 5 we have

$$\begin{aligned}\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\mathbf{p_c}+\triangle\mathbf{m}}(u,-\triangle\mathbf{m}),(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t)&\approx\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle m}(u,-\triangle\mathbf{m}),(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t)\\ &=\mathbf{W}_{\hat{\mathbf{p}}_{t-1}+\triangle\mathbf{m}}(u,(\mathbf{p_c}-\hat{\mathbf{p}}_{t-1})-\mathbf{m}_t-\triangle\mathbf{m})\approx\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{u},(\mathbf{p_c}-\hat{\mathbf{p}}_{\mathbf{t-1}})-\mathbf{m_t}-\triangle\mathbf{m}).\end{aligned} \tag{34}$$

Using the same reasoning as in (24)-(25), and under the assumption of neglecting second and higher order pose variations, (33) can be approximated as

$$\int_V (f(v, \mathbf{p}_c) - I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(v, (\mathbf{p}_c - \hat{\mathbf{p}}_{t-1}) - \mathbf{m}_t - \triangle\mathbf{m})))^2 dv, \tag{35}$$

which is equivalent to (29) with the additive update rule.

In a video sequence, $\mathbf{p}_c - \hat{\mathbf{p}}_{t-1}$ might become large as t increases. This invalidates the assumption used in deriving (33). Thus, the cardinal pose needs to be changed within a long sequence. In our experiments, we found that this reinitialization was needed for every $15^\circ - 20^\circ$. The physical interpretation of this is that the visibility of a significant portion of the object will change due to the difference between $\mathbf{p_c}$ and $\hat{\mathbf{p}}_{t-1}$, and thus $\mathbf{W}$ will no longer be reliable.

E. Overall Algorithm

Consider a sequence of image frames I_t, $t = 0, ..., N-1$.

Initialization: Take the first frame of the video sequence, register the 3D model onto this frame and map the texture onto the 3D model. Take this pose as cardinal pose $\mathbf{p_c}$. Pre-compute the $\mathcal{C}_{\bullet|\mathbf{p_c}}$ and $\mathcal{B}_{\bullet|p_c}$ at this pose. Assume that we know the pose and illumination estimates for frame $t-1$, i.e., $\hat{\mathbf{p}}_{t-1}$ and $\hat{\mathbf{l}}_{t-1}$.

• Step 1. For the new input frame $I_t(\mathbf{v})$, apply the pose transformation operator to get the pose normalized version of the frame $I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, \mathbf{p_c} - \hat{\mathbf{p}}_{t-1}))$. Let $\hat{\mathbf{l}}_t = \hat{\mathbf{l}}_{t-1}$, and $\hat{\mathbf{m}}_t = 0$.

• Step 2. Compute the increments of motion $\triangle\mathbf{m}$ using (32), and update the motion $\hat{\mathbf{m}}_t \leftarrow \hat{\mathbf{m}}_t + \triangle\mathbf{m}$.

• Step 3. Use $\hat{\mathbf{m}}_t$ to update the pose normalized image $I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, \mathbf{p_c} - \hat{\mathbf{p}}_{t-1} - \hat{\mathbf{m}}_t))$.

• Step 4. Use pre-computed $\mathcal{B}_{\bullet|p_c}$ and equation (4) to estimate the illumination vector $\hat{\mathbf{l}}_t$ of the updated pose normalized image $I_t(\mathbf{W}_{\hat{\mathbf{p}}_{t-1}}(\mathbf{v}, \mathbf{p_c} - \hat{\mathbf{p}}_{t-1} - \hat{\mathbf{m}}_t))$.

• Step 5. Repeat Steps 2, 3 and 4 with the new estimated $\hat{\mathbf{l}}_t$ for that input frame till the difference error between the input frame and the rendered frame can be reduced lower than an acceptable threshold.

• Step 6. If the $\hat{\mathbf{p}}_t - \mathbf{p_c}$ is larger than a threshold, re-initialize $\hat{\mathbf{p}}_t$ as the new cardinal pose $\mathbf{p_c}$. Re-compute $\mathcal{C}_{\bullet|\mathbf{p_c}}$ and $\mathcal{B}_{\bullet|p_c}$ at this new cardinal pose.

• Step 7. Set t = t + 1. Repeat Steps 1, 2, 3, 4, 5 and 6. Continue till t = N - 1.

F. Computational Complexity Analysis

The computation of $\mathcal{B}$ and $\mathcal{C}$ needs to exhaustively search over all the pixels, while the IC algorithm saves significant computational cost by pre-computing the derivatives $\mathcal{B}|_{\mathbf{p_c}}$ and $\mathcal{C}|_{\mathbf{p_c}}$ at the cardinal pose $\mathbf{p_c}$. In both approaches, a number of iterations will be needed to track each frame. As shown in section III-C and III-D, the increments $\triangle\mathbf{m}$ obtained from (32) in IC approach is approximately the same order as the $\triangle\mathbf{m}$ obtained from (14) in the direct approach, thus about the same number of iterations will be needed. In each iteration, the direct approach needs to compute the derivatives $\mathcal{B}_{\mathbf{v}|\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t}$ and $\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t}$, while the IC approach needs to compute the 3D warping $\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}(\mathbf{v}, \triangle\mathbf{p})$. According to the definition of $\mathcal{B}$ and $\mathcal{C}$ in [4], we need 24 multiplications plus 2 additions for computing $\mathcal{B}_{\mathbf{v}|\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t}$ and 93 multiplications plus 24 additions for computing $\mathcal{C}_{\mathbf{v}|\hat{\mathbf{p}}_{t-1}+\mathbf{m}_t}$ at only one pixel $\mathbf{v}$, while only one assignment operation (mapping the intensity at $I_t(\mathbf{v})$ to $I_t(\mathbf{W}_{\mathbf{p}}(\mathbf{v}, \triangle\mathbf{p}))$) will be needed for computing $\mathbf{W}_{\hat{\mathbf{p}}_{\mathbf{t-1}}}$ at the same pixel $\mathbf{v}$. Thus, by precomputing the the derivatives $\mathcal{B}|_{\mathbf{p_c}}$ and $\mathcal{C}|_{\mathbf{p_c}}$ at the cardinal pose $\mathbf{p_c}$, a significant amount of computation can be saved. The saving will depends upon the implementation, and our experimental results show that the IC algorithm has an average speed-up of > 50 times (maximum > 100) over the direct approach for the controlled data, while for the uncontrolled data, the average speed-up is over 30 times with maximum of 75.9 times, while maintaining the same estimation accuracy.

IV. Experimental Analysis of Computation Time and Accuracy

A. Analysis on Controlled Data

To show the tracking accuracy of the IC tracking algorithms, we first do a synthetic experiment with the Stanford Bunny rabbit model under varying illumination conditions. The bunny rabbit is rotating along the vertical axis at some specific angular velocity, and the illumination is changing both in direction (from right-bottom corner to the left-top corner) and in brightness (from dark to bright to dark). The first row in Fig. 2 shows the back projection of some feature points on the 3D model back onto the input frames using the estimated motion with the IC tracking algorithm under three different illumination conditions. The second row shows the synthesis images with the motion and illumination estimates. There is no perceptual difference between the original frame and the synthesized ones.

In Fig. 3, we compare the IC algorithm with the direct approach described in section II. We showed the comparison of the computational cost between the two approaches in (a), the motion estimation accuracy in (b), and the frequency of convergence in (c). The computational cost is measured by the processing time needed for each frame on a standard PC with 1.8GHz CPU, 2G RAM with a Matlab implementation. The average processing time for each frame in direct approach is 9.7 seconds, while in IC algorithm it is 0.18 seconds per frame. Thus, IC algorithm has an 52.1 fold speeding up while sacrificing little in the estimation accuracy. The frequency of convergence is computed as the percentage of the frames among the 180 frames in the control experiment that converge to the specific accuracy of the pose estimates measured in degree. On the average, the direct approach and the IC algorithm have the same frequency of convergence, validating the equivalence between the two. The divergence of the two curves is due to the relatively small number of the frames used for measuring the percentage of the convergence.

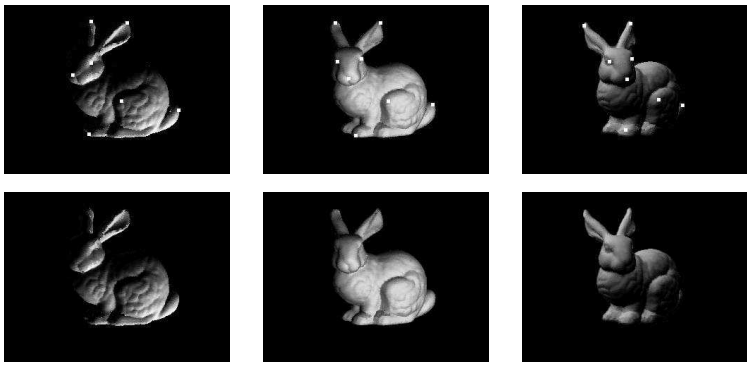
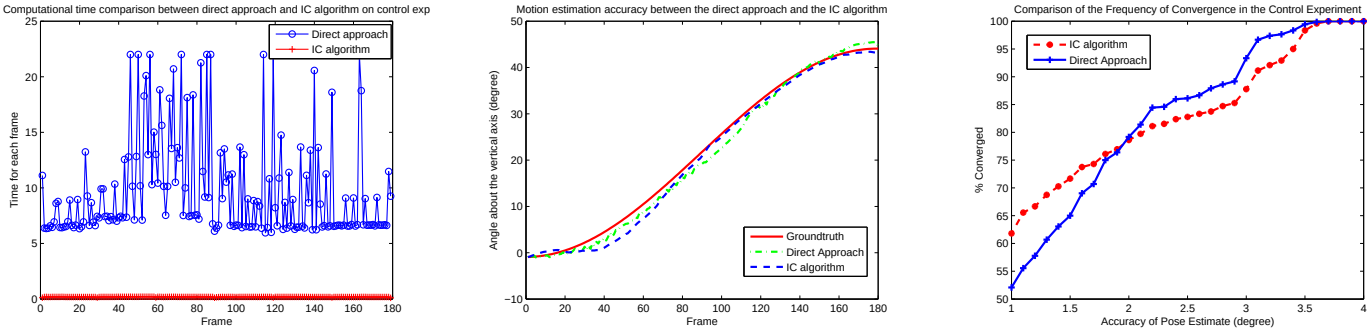


Fig. 2. Top: the back projection of the mesh vertices of the 3D bunny rabbit model using the estimated 3D motion onto some input frames. Bottom: Synthesized images with estimated motion and illumination.

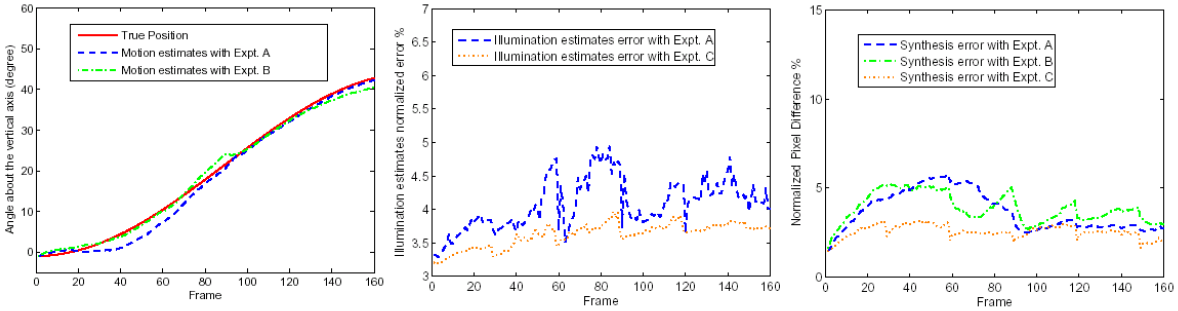
In Fig. 4, we show some accuracy analysis of the motion and illumination estimation. We designed three experiments: Expt. A - estimate both motion and illumination simultaneously; Expt. B - estimate motion with known illumination; Expt. C - estimate illumination with known motion. We show the results of this analysis in Fig. 4.

Note that illumination bases $\mathcal{B}$ are functions of pose, while the motion bases $\mathcal{C}$ do not rely upon illumination. Thus, knowing motion should be helpful for estimating the illumination. This is seen in Fig. 4 (b) where the illumination estimation error in Expt. C is consistently lower than that of Expt. A. Due to the same reason, the synthesis error in Expt. C is consistently lower than that in Expt. A, as shown in Fig. 4 (c). On the other hand, knowing illumination does not



(a) (b) (c)

Fig. 3. (a) shows the comparison of the computational time needed for each frame in the direct approach and the IC algorithm. (b) shows the comparison of the motion estimation accuracy obtained by the direct approach and the IC algorithm. (c) shows the comparison of the frequency of convergence in the control experiment between the direct approach and the IC algorithm.



(a) (b) (c)

Fig. 4. (a) shows the comparison between the pose estimates with known illumination, unknown illumination, and the ground truth, (b) shows the normalized error of the illumination estimates without knowing the true motion and with the true motion known, (c) shows the normalized synthesis error with unknown illumination and motion, unknown motion but known illumination, and unknown illumination but known motion.

help as much in motion estimation, since the motion bases do not depend upon illumination. Thus, the motion estimates of Expt. A are neither consistently better nor worse than those of Expt. B as shown in Fig. 4 (a), and the same is true for the synthesis errors, shown in Fig. 4 (c). Thus, knowing the ground truth motion can lead to more accurate estimates of illumination (the average synthesis error is 2.51%), while knowledge of illumination produces an average

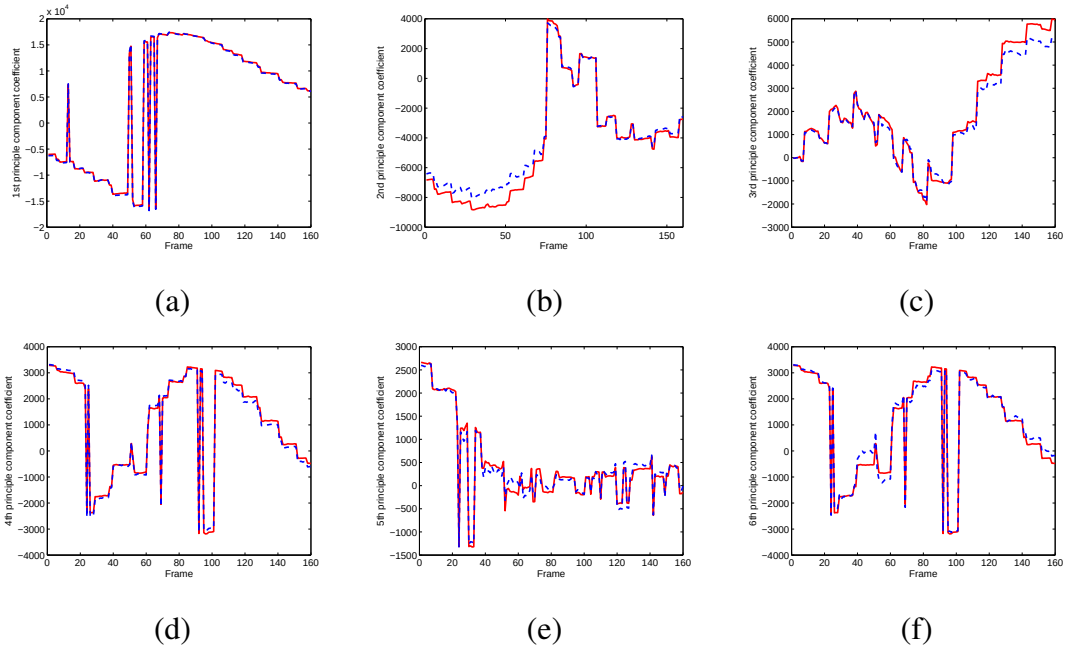


Fig. 5. (a) to (f) show the plots of the true and the estimated coefficients from the 1st to the 6th illumination principle components. The solid red plots are for the true illumination vector, the dotted blue ones are the illumination coefficients estimated from the inverse compositional algorithm.

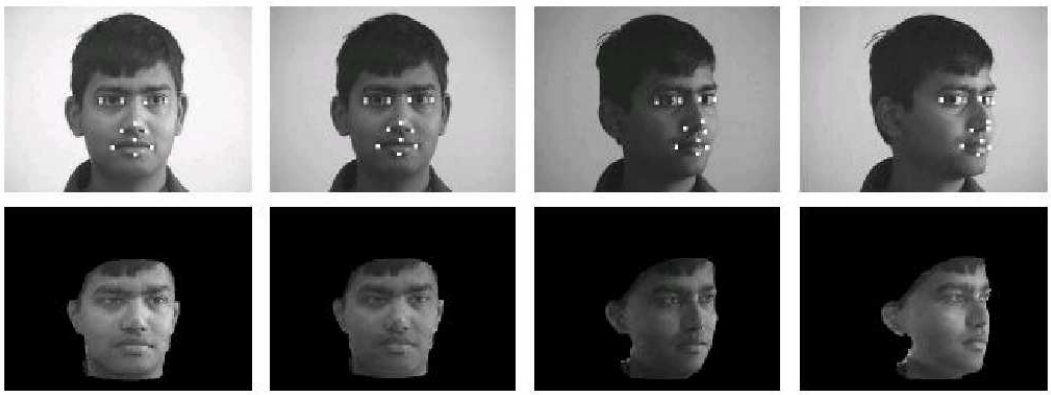


Fig. 6. The comparison between the original frames and the synthesized ones with the estimated motion and illumination variables. The first rows show the original frames, and the second row shows the synthesized frames with the estimated illumination and motion from the images in the same column.

synthesis error of 3.78%. In Fig. 5, we show the plots of six illumination coefficients [6].

[6]It has been shown that from a specific viewing point, the spherical harmonic functions will not be orthogonal to each other; therefore, not all the illumination coefficients will be observable [22]. We orthogonalize the spherical harmonic basis functions by taking their principal components, and estimate the illumination condition with this principal component basis.

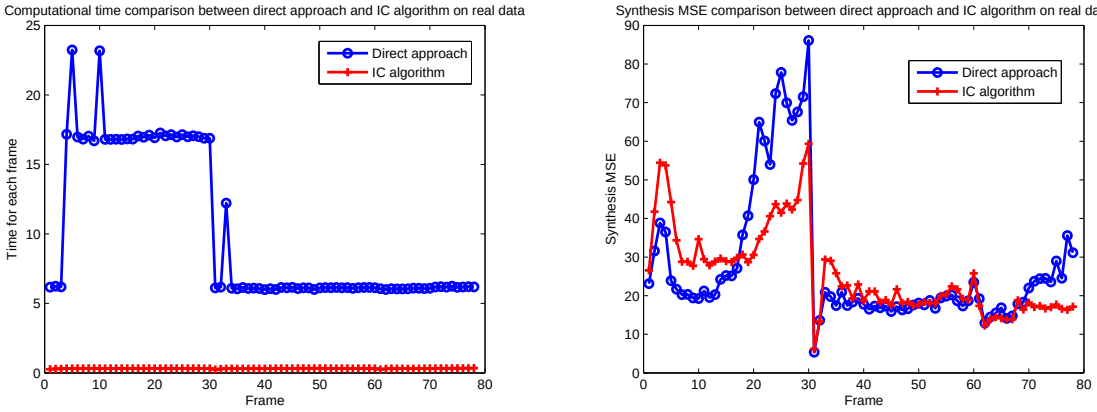
B. Analysis on Real-Life Face Data

Fig. 6 shows the motion and illumination estimates on two real data examples. The images in the first row are the input frames with the back projection of some feature mesh vertices, and the ones in the second row are synthesized with the estimated illumination and motion. This result shows that it is possible to synthesize images with the motion and illumination parameters learned from natural videos. This is extremely useful for applications in video-based rendering and object recognition.

In Fig. 7, we show the comparison of the computational cost and the estimation accuracy between the direct approach described in section II and the inverse compositional approach described in d) in section III on the sequence shown in the first row of Fig. 6. We use totally 80 frames, in which the head rotates from frontal pose to about 45 degree along the vertical axis. To assess the quality of the motion and illumination estimation accuracy on the real data, we synthesize the images with the estimated motion and illumination parameters, and take the pairwise pixel intensity difference between the synthesized frame and the input frame. Some peaks and plateaus in the plot of the direct approach in Fig. 7 (a) indicate that at those frames more iterations are needed for convergence. It is also shown in the plot that usually it takes about 6 seconds for one computation of the bilinear bases; thus the processing time for each frame is approximately a multiple of this time. From Fig. 7 (a), we can see that more iterations are used in the first 30 frames. This is because the motion between these frames is large and hence more iterations are needed. Around frame 30, the synthesis error was above a threshold and a new cardinal pose was chosen. After this, the inter-frame motion is smaller and the computation time and synthesis error are low. By taking the mean of the processing time for the 80 frames, the inverse compositional approach is 31.6 times faster than the direct approach, while the synthesis error is about the same in both approaches. The maximum improvement is 75.9 faster than the direct approach at specific frame. This shows the significant improvement of the IC algorithm over the direct approach.

V. CONCLUSIONS

In this paper, we presented an accurate and efficient approach for estimating illumination and 3D rigid motion from a video sequence. The work is based on a recently proposed theory that the set of images of a moving Lambertian object lies in a bilinear space of motion and illumination



(a) (b)

Fig. 7. Computational cost and the estimation accuracy comparison between the direct approach and the inverse compositional algorithm on the real data. (a) The vertical axis shows the processing time needed for each frame, while the horizontal axis shows the index of frames. By taking the mean of the processing time for all the frames in each approach, the direct approach has an average processing time of 10.11 seconds for each frame, while the IC algorithm uses 0.32 seconds per frame. (b) the vertical axis shows the MSE between the input frame and the synthesized frames using the estimated motion and illumination parameters.

parameters. We showed that it is possible to estimate the motion and lighting parameters using the inverse compositional approach. We proposed a new warping function, proved the converge of the IC approach, and showed experimental results on accuracy and computational efficiency. We presented experimental evaluation on controlled data with known ground truth, tracking results on real data and results in video synthesis.

REFERENCES

[1] G. D. Hager and P.N. Belhumeur, "Efficient Region Tracking With Parametric Models of Geometry and Illumination," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 10, pp. 1025–1039, 1998.

[2] D. Freedman and M. Turek, "Illumination-Invariant Tracking via Graph Cuts," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[3] H. Jin, P. Favaro, and S. Soatto, "Real-time feature tracking and outlier rejection with changes in illumination," in *IEEE Intl. Conf. on Computer Vision*, 2001.

[4] Y. Xu and A. Roy-Chowdhury, "Integrating Motion, Illumination and Structure in Video Sequences, With Applications in Illumination-Invariant Tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 793–807, May 2007.

[5] R. Basri and D.W. Jacobs, "Lambertian Reflectance and Linear Subspaces," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218–233, February 2003.

[6] R. Ramamoorthi and P. Hanrahan, "On the relationship between radiance and irradiance: determining the illumination from images of a convex Lambertian object," *Journal of the Optical Society of America A*, vol. 18, no. 10, Oct 2001.

[7] S. Baker and I. Matthews, "Lucas-Kanade 20 Years On: A Unifying Framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, Mar. 2004.

[8] B. D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision (DARPA)," in *Proceedings of the 1981 DARPA Image Understanding Workshop*, April 1981.

[9] H.-Y. Shum and R. Szeliski, "Construction of panoramic image mosaics with global and local alignment," *International Journal of Computer Vision*, vol. 16, no. 1, pp. 63–84, 2000.

[10] I. Matthews and S. Baker, "Active Appearance Models Revisited," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 135–164, Nov. 2004.

[11] Sami Romdhani and Thomas Vetter, "Efficient, robust and accurate fitting of a 3d morphable model," in *IEEE International conference on Computer Vision 2003*, 2003.

[12] A. Bartoli, "Groupwise geometric and photometric direct image registration.," in *BMVC06. 7th BRITISH MACHINE VISION CONFERENCE, EDINBURGH, UK*, Sep 2006.

[13] V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1063–1074, September 2003.

[14] P. Eisert and B. Girod, "Illumination compensated motion estimation for analysis sythesis coding," *3D Image Analysis and Synthesis*, pages 61-66, 1996.

[15] M. Gouiffes, C. Collewet, C. Fernandez-Maloigne, and A. Trémeau, "Feature points tracking : robustness to specular highlights and lighting changes.," in *ECCV06. 9th European Conference on Computer Vision, Graz, Austria*, May 2006.

[16] G. Finlayson, S. Hordley, and M. Drew, "Removing shadows from images," 2002.

[17] D. Pizarro and A. Bartoli, "Shadow resistant direct image registration.," in *SCIA'07 - Proceedings of the Fifteenth Scandinavian Conference on Image Analysis, Aalborg, Denmark*, Jun 2007.

[18] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[19] A. Kale and C. Jaynes, "A joint illumination and shape model for visual tracking," *Proceedings of IEEE CVPR*, pp. 602–609, 2006.

[20] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A Multillinear Singular Value Decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[21] R. Szeliski and S. B. Kang, "Recovering 3D shape and motion from image streams using non-linear least squares," *Journal of Visual Communication and Image Representation*, vol. 5, no. 1, pp. 10–28, 1994.

[22] R. Ramamoorthi, "Analytic PCA Construction for Theoretical Analysis of Lighting Variability in Images of a Lambertian Object," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2002.