Dynamic 2D/3D Registration for the Kinect

Sofien Bouaziz Mark Pauly
École Polytechnique Fédérale de Lausanne

Abstract

Image and geometry registration algorithms are an essential component of many computer graphics and computer vision systems. With recent technological advances in RGB-D sensors, such as the Microsoft Kinect or Asus Xtion Live, robust algorithms that combine 2D image and 3D geometry registration have become an active area of research. The goal of this course is to introduce the basics of 2D/3D registration algorithms and to provide theoretical explanations and practical tools to design computer vision and computer graphics systems based on RGB-D devices. To illustrate the theory and demonstrate practical relevance, we briefly discuss three applications: rigid scanning, non-rigid modeling, and realtime face tracking. Our course targets researchers and computer graphics practitioners with a background in computer graphics and/or computer vision.

About the Lecturers

Sofien Bouaziz is a PhD student in the Computer Graphics and Geometry Laboratory at the École Polytechnique Fédérale de Lausanne (EPFL) under the supervision of Prof. Mark Pauly. He received his MSc degree in Computer Science from EPFL in 2009. His research interests include computer graphics, computer vision, and machine learning. Sofien co-developed the facial motion capture software *faceshift studio*.

Mark Pauly is an associate professor of computer science at EPFL in Lausanne, Switzerland, where he directs the Computer Graphics and Geometry Laboratory. Prior to joining EPFL he was an assistant professor at ETH Zurich and a postdoctoral scholar at Stanford University. He received his Ph.D. degree in 2003 from ETH Zurich. His research interests include computer graphics and animation, shape analysis, geometry processing, and architectural design.

Sofien and Mark are co-founders of faceshift AG (www.faceshift.com), an EPFL spin-off that brings high-quality markerless facial motion capture to the consumer market.

Course Overview

5 minutes: Introduction

- Overview of the course and motivation

50 minutes: 2D/3D Registration

- 20 minutes: 3D Registration
- 20 minutes: 2D Registration
- 10 minutes: Putting It All Together

30 minutes: Applications

- 10 minutes: Rigid Scanning
- 10 minutes: Non-rigid Modeling
- 10 minutes: Realtime Face Tracking

5 minutes: Conclusion

- Outlook and Q&A

1 Introduction

Recent technological advances in RGB-D sensing devices, such as the Microsoft Kinect, facilitate numerous new and exciting applications, for example in 3D scanning [19] and human motion tracking [20, 15, 5]. While affordable and accessible, consumer-level RGB-D devices typically exhibit high noise levels in the acquired data. Moreover, difficult lighting situations and geometric occlusions commonly occur in many application settings, potentially leading to a severe degradation in data quality. This necessitates a particular emphasis on the robustness of image and geometry processing algorithms. The combination of 2D and 3D registration is one important aspect in the design of robust applications based on RGB-D devices. This lecture introduces the main concepts of 2D and 3D registration and explains how to combine them efficiently. An up-to-date version of these course notes as well as implementation details and source code can be found at `http://lgg.epfl.ch/2d3dRegistration`.

2 2D/3D Registration

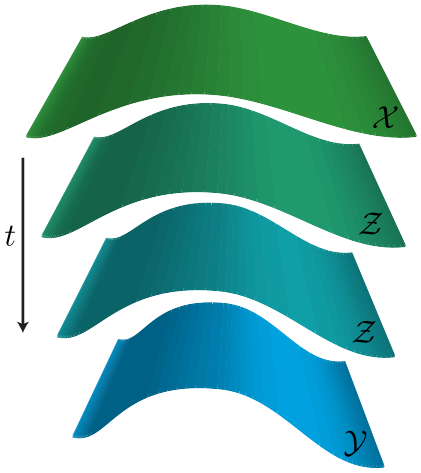
In the first part of the course we introduce the theory of 2D/3D registration algorithms suitable for processing RGB-D data. We focus on pairwise registration to compute the alignment of a source model onto a target model. This alignment can be rigid or non-rigid, depending on the type of object being scanned. We formulate the registration as the minimization of an energy

$$E_{\text{reg}} = E_{\text{match}} + E_{\text{prior}}. \tag{1}$$

The matching energy E_{match} defines a measure of how close the source is from the target. The prior energy E_{prior} quantifies the deviation from the type of transformation or deformation that the source is allowed to undergo during the registration, for example, a rigid motion or an elastic deformation. The goal of registration is to find a transformation of the source model that minimizes E_{reg} to bring the source into alignment with the target. For data acquired with RGB-D devices, registration can utilize both the geometric information encoded in the 3D depth map, as well as the color information provided by the recorded 2D images. We show that Equation 1 provides a unified way to formulate both 2D and 3D registration, which simplifies their integration.

2.1 3D Registration

In 3D registration we want to align a source surface $\mathcal{X}$ embedded in $\mathbb{R}^3$ to a target surface $\mathcal{Y}$ in $\mathbb{R}^3$. To formalize this problem, we introduce a surface $\mathcal{Z}$ that is a transformed or deformed version of $\mathcal{X}$ that eventually aligns with $\mathcal{Y}$. To solve the registration problem numerically, we represent the continuous surface $\mathcal{X}$ by a set of points $X = \{\mathbf{x}_i \in \mathcal{X}, i = 1 \dots n\}$ and define their corresponding points on the deformed surface $\mathcal{Z}$ as $Z = \{\mathbf{z}_i \in \mathcal{Z}, i = 1 \dots n\}$. Different sampling strategies have been presented by Rusinkiewicz and Levoy [16].



2.1.1 Matching Energy

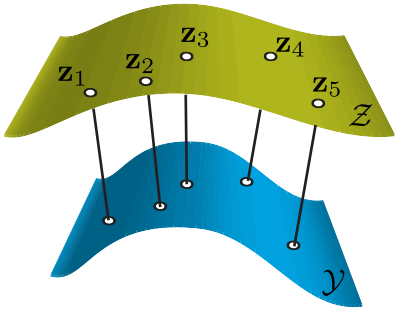
The matching energy measures how close the surface $\mathcal{Z}$ is to the surface $\mathcal{Y}$ and is defined as

$$E_{\text{match}}(\mathcal{Z}) = \int_{\mathcal{Z}} \varphi(\mathbf{z}, \mathcal{Y}) d\mathbf{z}, \tag{2}$$

where $\mathbf{z} \in \mathbb{R}^3$ is a point on $\mathcal{Z}$. The accuracy of the registration is evaluated by the metric φ that measures the distance to $\mathcal{Y}$. For simplicity, we use the squared Euclidian distance as metric for all the energies presented in this course. However, robust metrics [13] could be use instead to increase the robustness of the registration to noise and outliers. Using the set of points Z, we can discretize the matching energy as

$$E_{\text{match}}(\mathcal{Z}) = \sum_{i=1}^{n} \|\mathbf{z}_i - P_{\mathcal{Y}}(\mathbf{z}_i)\|_2^2. \tag{3}$$

where $P_{\mathcal{Y}}(\mathbf{z}_i)$ is the closest point (using Euclidian distance) on the surface $\mathcal{Y}$ from $\mathbf{z}_i$. $P_{\mathcal{Y}}(\mathbf{z}_i)$ can also be seen as the orthogonal projection of $\mathbf{z}_i$ onto $\mathcal{Y}$.



2.1.2 Prior Energy

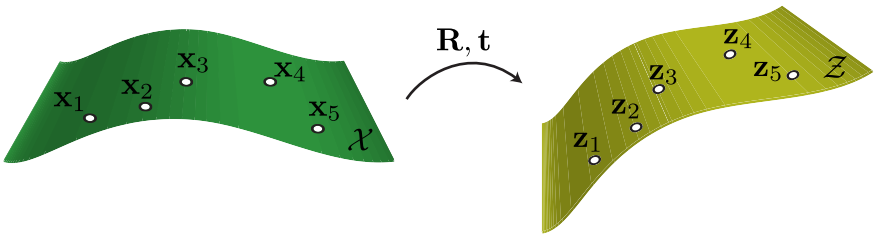
In this section we present several prior energies that can be used for registration. These energies can also be combined to build more sophisticated priors. Priors encode properties of the scanned objects. For example, when scanning rigid objects, a global rigidity

prior can be used to limit the allowed transformations to rotations and translations. For deforming objects, for example a human body, geometric priors are often employed that try to mimic physical behavior such as an elastic deformation. We describe a simple local rigidity prior that approximates elastic deformations and facilitates efficient implementations. More complex deformation behavior can be captured using a data-driven approach. One popular method is based on a collection of sample shapes that represent that allowable space of deformations. Using dimensionality reduction, for example principal component analysis, efficient linear models can be derived that are suitable for realtime registration algorithms.

Global Rigidity. The global rigidity of the 3D registration can be measured as

$$E_{\text{prior}}(Z, \mathbf{R}, \mathbf{t}) = \sum_{i=1}^{n} \|\mathbf{z}_i - (\mathbf{R}\mathbf{x}_i + \mathbf{t})\|_2^2, \tag{4}$$

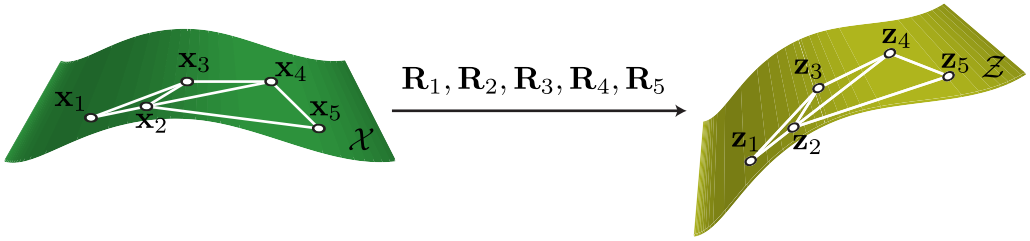
where $\mathbf{R} \in \mathbb{R}^{3\times3}$ is a rotation matrix and $\mathbf{t} \in \mathbb{R}^3$ a translation vector. In this case, the surface $\mathcal{Z}$ tries to follow a rigid transformation of the surface $\mathcal{X}$.



Local Rigidity. The local rigidity energy, following [17, 4], can be expressed as

$$E_{\text{prior}}(Z, \mathbf{R}_i|_{i=1}^{n}) = \sum_{i=1}^{n} \sum_{j\in\mathcal{N}_i} \|(\mathbf{z}_j - \mathbf{z}_i) - \mathbf{R}_i(\mathbf{x}_j - \mathbf{x}_i)\|_2^2, \tag{5}$$

where the $\mathbf{R}_i \in \mathbb{R}^{3\times3}$ are rotation matrices and $\mathcal{N}_i$ is the set of indices of the neighboring points of $\mathbf{x}_i$. In this case, each local neighborhood on the surface $\mathcal{Z}$ tries to follow a rigid transformation of its corresponding local neighborhood on the surface $\mathcal{X}$. Other *local rigidity* energies can also be used as prior, see for example [3, 18].



Linear Model. A 3D linear shape model can be defined using a matrix $\mathbf{P}$ containing the shape model basis, and a mean shape vector $\mathbf{m}$ [7]. A new shape $\mathbf{s}$ can be defined as

$$\mathbf{s} = \mathbf{P}\mathbf{d} + \mathbf{m}, \tag{6}$$

where $\mathbf{d}$ is a vector containing the basis coefficients. A linear model prior energy can be formulated as the deviation of the vertices from the linear model

$$E_{\text{prior}}(Z, \mathbf{d}) = \sum_{i=1}^{n} \|\mathbf{z}_i - (\mathbf{P}_i\mathbf{d} + \mathbf{m}_i)\|_2^2, \tag{7}$$

where $\mathbf{P}_i$ and $\mathbf{m}_i$ are the part of $\mathbf{P}$ and $\mathbf{m}$ corresponding to the vertex $\mathbf{z}_i$.

2.1.3 Optimization

How to best optimize the registration energy depends on the prior energy. In this section we show, as an example, how to optimize a registration energy for two applications: rigid scanning and non-rigid modeling.

Rigid Scanning Since single depth maps acquired with the RGB-D sensor exhibit high noise levels and do not cover the whole surface of the 3D object, an aggregation procedure is typically applied to obtain a complete model with reduced noise level. In order to aggregate multiple scans over time, different methods can be used [22, 23, 14]. The classical approach is to perform a 3D rigid registration of the currently acquired scan of the object with the already accumulated 3D data. The pairwise 3D alignment can be formulated as

$$E_{\text{reg}}(Z, \mathbf{R}, \mathbf{t}) = w_1 \sum_{i=1}^{n} \|\mathbf{z}_i - P_{\mathcal{Y}}(\mathbf{z}_i)\|_2^2 + w_2 \sum_{i=1}^{n} \|\mathbf{z}_i - (\mathbf{R}\mathbf{x}_i + \mathbf{t})\|_2^2, \tag{8}$$

where the matching energy is combined with a global rigidity prior. To optimize $E_{\text{reg}}(Z, \mathbf{R}, \mathbf{t})$ we linearize the rotation matrix approximating $\cos\theta$ by 1 and $\sin\theta$ by θ

$$\mathbf{R} \approx \tilde{\mathbf{R}} = \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}. \tag{9}$$

The alignment is computed by solving iteratively

$$\underset{Z^{t+1}, \mathbf{R}, \mathbf{t}}{\arg\min}\, w_1 \sum_{i=1}^{n} \|\mathbf{z}_i^{t+1} - P_{\mathcal{Y}}(\mathbf{z}_i^t)\|_2^2 + w_2 \sum_{i=1}^{n} ||\mathbf{z}_i^{t+1} - (\tilde{\mathbf{R}}\mathbf{x}_i + \mathbf{t})||_2^2, \tag{10}$$

where t is the iteration number and $\mathbf{z}_i^0 = \mathbf{x}_i$. As $P_{\mathcal{Y}}(.)$ is a non linear function that is difficult to optimize with, we use in the optimization the previous estimate $P_{\mathcal{Y}}(\mathbf{z}_i^t)$. This correspond to the point-to-point matching error [1]. To speed up the convergence of the optimization one can linearize $\|\mathbf{z}_i^{t+1} - P_{\mathcal{Y}}(\mathbf{z}_i^t)\|_2$ at $P_{\mathcal{Y}}(\mathbf{z}_i^t)$ which gives $\mathbf{n}_i^T(\mathbf{z}_i^{t+1} - P_{\mathcal{Y}}(\mathbf{z}_i^t))$,

where $\mathbf{n}_i$ is the normal of the surface $\mathcal{Y}$ at $P_{\mathcal{Y}}(\mathbf{z}_i^t)$. This leads to the point-to-plane matching error [6]. The optimization can be reformulated as

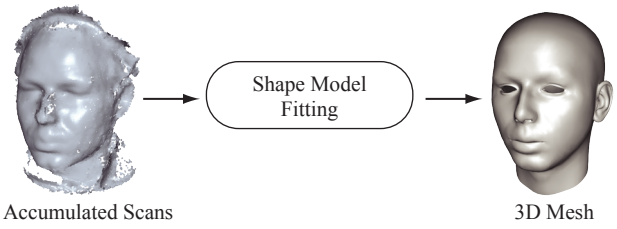
$$\underset{Z^{t+1},\mathbf{R},\mathbf{t}}{\arg\min}\, w_1 \sum_{i=1}^{n} (\mathbf{n}_i^T(\mathbf{z}_i^{t+1} - P_{\mathcal{Y}}(\mathbf{z}_i^t)))^2 + w_2 \sum_{i=1}^{n} ||\mathbf{z}_i^{t+1} - (\tilde{\mathbf{R}}\mathbf{x}_i + \mathbf{t})||_2^2. \tag{11}$$

Both Equation 10 and Equation 11 are quadratic, and therefore, can be optimized by setting the partial derivatives to zero by solving a linear system. It is interesting to note that if $w_2 = +\infty$ then $\mathbf{z}_i$ can be replaced into the matching energy by $\mathbf{R}\mathbf{x}_i + \mathbf{t}$ leading to a registration energy

$$E_{\text{reg}}(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^{n} \|\mathbf{R}\mathbf{x}_i + \mathbf{t} - P_{\mathcal{Y}}(\mathbf{R}\mathbf{x}_i + \mathbf{t})\|_2^2. \tag{12}$$

This energy can be minimized in a similar spirit by linearizing the rotation matrix and iteratively solving a linear system. Other approaches can be found in [8].

Non-rigid Modeling Registering a shape template towards a scanned 3D object allows to obtain a complete and clean 3D mesh [11]. An example is given below in the context of face modeling. In this case, the morphable model of Blanz and Vetter [2] that represents the variations of different human faces in neutral expression was registered to a scan of a face. Non-rigid modeling using a morphable model can be formulated as



$$E_{\text{reg}}(Z, \mathbf{d}, \mathbf{R}_i|_{i=1}^n, \mathbf{R}, \mathbf{t}) = w_1 \sum_{i=1}^{n} \|\mathbf{z}_i - P_{\mathcal{Y}}(\mathbf{z}_i)\|_2^2 + w_2 \sum_{i=1}^{n} \|\mathbf{z}_i - (\mathbf{R}\mathbf{x}_i + \mathbf{t})\|_2^2 +$$
$$w_3 \sum_{i=1}^{n} \|\mathbf{z}_i - (\mathbf{P}_i\mathbf{d} + \mathbf{m}_i)\|_2^2 + w_4 \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \|(\mathbf{z}_j - \mathbf{z}_i) - \mathbf{R}_i(\mathbf{x}_j - \mathbf{x}_i)\|_2^2. \tag{13}$$

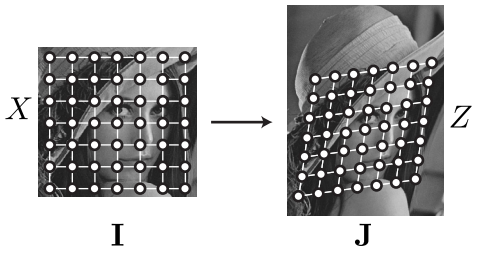
A local rigidity energy is added to the optimization in order to get an accurate result, as the morphable model represents the large-scale variability but might not capture small scale details. As previously, we solve iteratively

$$\underset{Z^{t+1},\mathbf{d},\mathbf{R}_i|_{i=1}^n,\mathbf{R},\mathbf{t}}{\arg\min}\, w_1 \sum_{i=1}^{n} (\mathbf{n}_i^T(\mathbf{z}_i^{t+1} - P_{\mathcal{Y}}(\mathbf{z}_i^t)))^2 + w_2 \sum_{i=1}^{n} ||\mathbf{z}_i^{t+1} - (\tilde{\mathbf{R}}\mathbf{x}_i + \mathbf{t})||_2^2 +$$
$$w_3 \sum_{i=1}^{n} \|\mathbf{z}_i^{t+1} - (\mathbf{P}_i\mathbf{d} + \mathbf{m}_i)\|_2^2 + w_4 \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \|(\mathbf{z}_j^{t+1} - \mathbf{z}_i^{t+1}) - \tilde{\mathbf{R}}_i(\mathbf{x}_j - \mathbf{x}_i)\|_2^2, \tag{14}$$

which corresponds to solving a linear system.

2.2 2D Registration

In 2D registration we want to register a source image $\mathbf{I}$ to a target image $\mathbf{J}$. During the registration process, the 2D pixel grid of the source image $X = \{\mathbf{x}_i \in \mathbb{R}^2, i = 1 \dots n\}$ is deformed to $Z = \{\mathbf{z}_i \in \mathbb{R}^2, i = 1 \dots n\}$ to match the target image.



2.2.1 Matching Energy

We define $\mathbf{I}(\mathbf{x})$ as the pixel value of the image $\mathbf{I}$ located at the position $\mathbf{x}$. The matching energy measures the color similarity between the source image and the target image wrapped onto the deformed grid Z.

$$E_{\text{match}}(Z) = \sum_{i=1}^{n} \|\mathbf{I}(\mathbf{x}_i) - \mathbf{J}(\mathbf{z}_i)\|_2^2. \tag{15}$$

2.2.2 Prior Energy

Similar to the 3D geometry registration, we can use different prior energies that can be combined to build more complex priors.

First Order Smoothness. In the Lucas-Kanade algorithm [12] the deformation is assumed to be constant within a patch around each pixel. This corresponds to the prior energy

$$E_{\text{prior}}(Z) = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_i} \|(\mathbf{z}_j - \mathbf{x}_j) - (\mathbf{z}_i - \mathbf{x}_i)\|_2^2, \tag{16}$$

where $\mathcal{N}_i$ is the set of indices of the neighbors of $\mathbf{x}_i$.

Laplacian Smoothness. In the Horn-Schunck algorithm [10] the smoothness of the flow is defined using a Laplacian operator

$$E_{\text{prior}}(Z) = \sum_{i=1}^{n} \|(\mathbf{z}_i - \mathbf{x}_i) - \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (\mathbf{z}_j - \mathbf{x}_j)\|_2^2, \tag{17}$$

where $|\mathcal{N}_i|$ is the cardinality of $\mathcal{N}_i$. This energy measures for each grid vertex the deviation of its deformation from the mean deformation of its neighbors.

2.2.3 Optimization

In this section we show, as an example, how to optimize the matching energy combined with the laplacian smoothness energy. This is similar to the method presented in [10]. The optimization is defined as

$$Z^* = \arg\min_Z E_{\text{reg}}(Z) \tag{18}$$

with

$$E_{\text{reg}}(Z) = w_1 \sum_{i=1}^{n} \|\mathbf{I}(\mathbf{x}_i) - \mathbf{J}(\mathbf{z}_i)\|_2^2 + w_2 \sum_{i=1}^{n} \|(\mathbf{z}_i - \mathbf{x}_i) - \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (\mathbf{z}_j - \mathbf{x}_j)\|_2^2. \tag{19}$$

To solve this optimization we linearize $\mathbf{J}(.)$ at the current estimate and solve iteratively

$$\begin{aligned} \arg\min_{Z^{t+1}} w_1 \sum_{i=1}^{n} \|\mathbf{I}(\mathbf{x}_i) - \mathbf{J}(\mathbf{z}_i^t) - \nabla\mathbf{J}(\mathbf{z}_i^t)^T(\mathbf{z}_i^{t+1} - \mathbf{z}_i^t)\|_2^2 + \\ w_2 \sum_{i=1}^{n} \|(\mathbf{z}_i^{t+1} - \mathbf{x}_i) - \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (\mathbf{z}_i^{t+1} - \mathbf{x}_i)\|_2^2. \end{aligned} \tag{20}$$

where $\nabla\mathbf{J} = \begin{bmatrix} \nabla\mathbf{J}_x & \nabla\mathbf{J}_y \end{bmatrix}^T$ is the image gradient, with $\nabla\mathbf{J}_x$ the image gradient in x direction and $\nabla\mathbf{J}_y$ the image gradient in y direction. As previously, the minimization can be computed by setting the partial derivative to zero, which corresponds to solving a linear system.

2.3 Putting It All Together

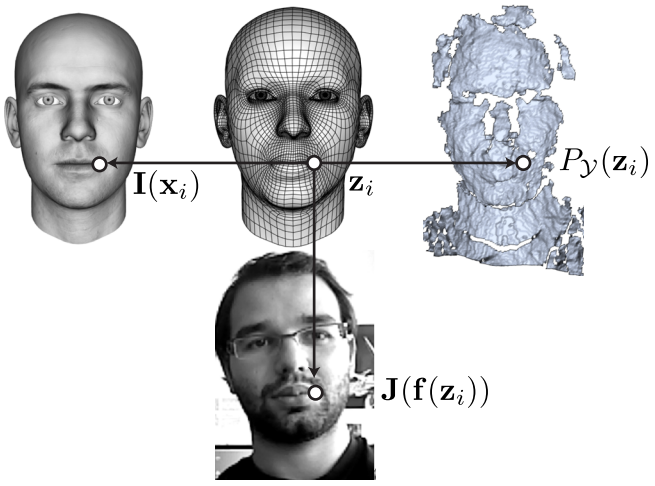
We show how to combine 2D image registration and 3D geometry registration to best utilize the data provided by the RGB-D sensor. More specifically, we want to register a surface $\mathcal{X} \subset \mathbb{R}^3$ with color information $\mathbf{I}$, i.e. a texture mapped surface, to a 3D surface $\mathcal{Y}$ with corresponding color image $\mathbf{J}$. As previously, the source $\mathcal{X}$ is deformed to a surface $\mathcal{Z}$. We sample the continuous surface $\mathcal{X}$ to obtain a set of points $X = \{\mathbf{x}_i \in \mathcal{X}, i = 1 \dots n\}$. We define their corresponding points on the deformed surface $\mathcal{Z}$ as $Z = \{\mathbf{z}_i \in \mathcal{Z}, i = 1 \dots n\}$. The color information of sample point $\mathbf{x}_i$ is given by $\mathbf{I}(\mathbf{x}_i)$.

2.3.1 Matching Energy

We formulate the energy measuring the quality of the 2D and 3D alignment as follow

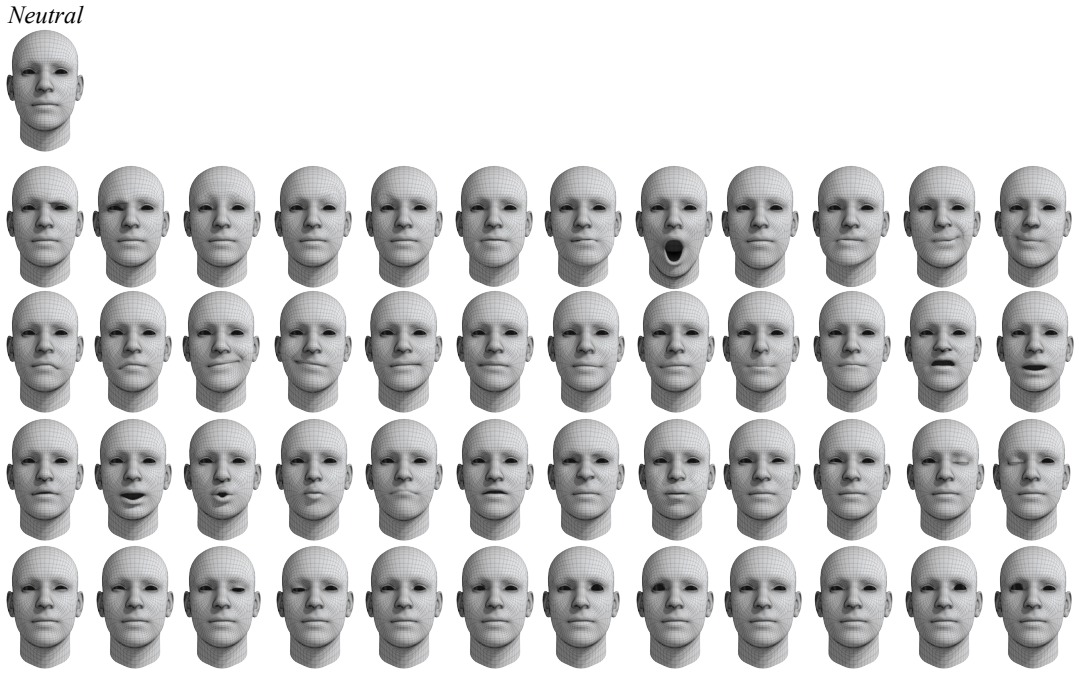
$$E_{\text{match}}(Z) = w_1 \sum_{i=1}^{n} \|\mathbf{z}_i - P_{\mathcal{Y}}(\mathbf{z}_i)\|_2^2 + w_2 \sum_{i=1}^{n} \|\mathbf{I}(\mathbf{x}_i) - \mathbf{J}(\mathbf{f}(\mathbf{z}_i))\|_2^2. \tag{21}$$

The first term is the matching energy presented in Section 2.1. The second term is similar to the 2D matching energy presented in Section 2.2. The only difference is the additional function $\mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^2$ that projects a 3D point $\mathbf{z}_i$ to the 2D image $\mathbf{J}$. For example this function could be a perspective projection of the form $\mathbf{f}(\mathbf{z}_i) = \begin{bmatrix} \frac{f\mathbf{z}_{i,x}}{\mathbf{z}_{i,z}} & \frac{f\mathbf{z}_{i,y}}{\mathbf{z}_{i,z}} \end{bmatrix}^T$.



2.3.2 Optimization

We illustrate 2D/3D registration in the context of a face tracking system that combines the 2D/3D matching energy with a 3D blendshape prior. A blendshape representation is a linear model defined as a set of blendshape meshes $\mathbf{B} = [\mathbf{b}^0, ..., \mathbf{b}^n]$ where $\mathbf{b}_0$ is the rest pose and $\mathbf{b}_i, i > 0$ are different expressions. A new expression can be generated as $\mathbf{T} = \mathbf{b}^0 + \mathbf{Bd}$, where $\mathbf{B} = [\mathbf{b}^1 - \mathbf{b}^0, ..., \mathbf{b}^n - \mathbf{b}^0]$. The blendshape model shown below is inspired from Ekmans Facial Action Coding System [9]. Realtime face tracking using an



RGB-D device can be formulated as a 2D/3D registration of the blendshape model to the 2D and 3D data [21]. The registration energy can be formulated as

$$E_{\text{reg}}(Z, \mathbf{d}, \mathbf{R}, \mathbf{t}) = w_1 \sum_{i=1}^{n} \|\mathbf{z}_i - P_{\mathcal{Y}}(\mathbf{z}_i)\|_2^2 + w_2 \sum_{i=1}^{n} \|\mathbf{I}(\mathbf{x}_i) - \mathbf{J}(\mathbf{f}(\mathbf{z}_i))\|_2^2 +$$
$$w_3 \sum_{i=1}^{n} ||\mathbf{z}_i - (\mathbf{B}_i\mathbf{d} + \mathbf{b}_i^0)||_2^2 + w_4 \sum_{i=1}^{n} \|\mathbf{z}_i - (\mathbf{R}\mathbf{x}_i + \mathbf{t})\|_2^2, \tag{22}$$

To solve this optimization we linearize $\mathbf{J}(\mathbf{f}(.))$ at the current estimate and solve iteratively

$$\arg\min_{Z^{t+1}, \mathbf{d}} w_1 \sum_{i=1}^{n} (\mathbf{n}_i^T(\mathbf{z}_i^{t+1} - P_{\mathcal{Y}}(\mathbf{z}_i^t)))^2 +$$
$$w_2 \sum_{i=1}^{n} \|\mathbf{I}(\mathbf{x}_i) - \mathbf{J}(\mathbf{f}(\mathbf{z}_i^t)) + \nabla\mathbf{J}(\mathbf{f}(\mathbf{z}_i^t))^T \frac{\partial f(\mathbf{z}_i^t)}{\partial \mathbf{z}}(\mathbf{z}^{t+1} - \mathbf{z}^t))\|_2^2 +$$
$$w_3 \sum_{i=1}^{n} ||\mathbf{z}_i^{t+1} - (\mathbf{B}_i\mathbf{d} + \mathbf{b}_i^0)||_2^2 + + w_4 \sum_{i=1}^{n} \|\mathbf{z}_i^{t+1} - (\tilde{\mathbf{R}}\mathbf{x}_i + \mathbf{t})\|_2^2. \tag{23}$$

For a perspective projection $\mathbf{f}(\mathbf{z}_i) = \begin{bmatrix} \frac{f\mathbf{z}_{i,x}}{\mathbf{z}_{i,z}} & \frac{f\mathbf{z}_{i,y}}{\mathbf{z}_{i,z}} \end{bmatrix}^T$ we have

$$\frac{\partial f(\mathbf{z}_i)}{\partial \mathbf{z}} = \begin{bmatrix} \frac{f}{\mathbf{z}_{i,z}} & 0 & -\frac{f\mathbf{z}_{i,x}}{\mathbf{z}_{i,z}^2} \\ 0 & \frac{f}{\mathbf{z}_{i,z}} & -\frac{f\mathbf{z}_{i,y}}{\mathbf{z}_{i,z}^2} \end{bmatrix}. \tag{24}$$

As previously, the minimization can be computed by solving a linear system. For tracking, another 2D matching energy can also be added

$$E_{\text{match}}(Z^{t+1}) = \sum_{i=1}^{n} \|\mathbf{J}_t(\mathbf{f}(\mathbf{z}_i^t)) - \mathbf{J}_{t+1}(\mathbf{f}(\mathbf{z}_i^{t+1}))\|_2^2. \tag{25}$$

This energy enforces color consistency over time by measuring the variation of color from the previous image frame $\mathbf{J}_t$ to the current frame $\mathbf{J}_{t+1}$ for each $\mathbf{z}_i$. In [21], the global rigidity is decoupled. In a first step, a 3D rigid alignment is performed, and in second step, a 2D/3D alignment of the blendshape model is computed.

3 Conclusion

We have shown that 2D and 3D registration can be expressed and combined in a common framework. Numerous application based on RGB-D devices can benefit from this formulation that allows to combine different priors in an easy manner.

References

[1] P. Besl and H. McKay. A method for registration of 3d shapes. *PAMI*, 1992.

[2] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. Proc. of ACM SIGGRAPH, 1999.

[3] M. Botsch, M. Pauly, M. Gross, and L. Kobbelt. Primo: coupled prisms for intuitive surface modeling. SGP, 2006.

[4] S. Bouaziz, M. Deuss, Y. Schwartzburg, T. Weise, and M. Pauly. Shape-up: Shaping discrete geometry with projections. *Comput. Graph. Forum*, 2012.

[5] S. Bouaziz, Y. Wang, and M. Pauly. Online modeling for realtime facial animation. *ACM Trans. Graph.*, 2013.

[6] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *ICRA*, 1991.

[7] T. Cootes and C. Taylor. Statistical models of appearance for computer vision, 2000.

[8] D. W. Eggert, A. Lorusso, , and R. B. Fisher. Estimating 3-d rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications*, 1997.

[9] P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Consulting Psychologists Press, 1978.

[10] B. K. P. Horn and B. G. Schunck. "determining optical flow": A retrospective. *Artif. Intell.*, 1993.

[11] H. Li, B. Adams, L. J. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. *ACM Trans. Graph.*

[12] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. IJCAI, 1981.

[13] M. Mirza and K. Boyer. Performance evaluation of a class of m-estimators for surface parameter estimation in noisy range data. *IEEE Transactions on Robotics and Automation*, 1993.

[14] 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. ISMAR, 2011.

[15] I. Oikonomidis, N. Kyriazis, and A. Argyros. Tracking the articulated motion of two strongly interacting hands. CVPR, 2012.

[16] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. *3DIM*, 2001.

[17] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. SGP, 2007.

[18] R. W. Sumner, J. Schmid, and M. Pauly. Embedded deformation for shape manipulation. *ACM Trans. Graph.*, 2007.

[19] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan. Scanning 3d full human bodies using kinects. *TVCG*, 2012.

[20] X. Wei, P. Zhang, and J. Chai. Accurate realtime full-body motion capture using a single depth camera. *ACM Trans. Graph.*, 2012.

[21] T. Weise, S. Bouaziz, H. Li, and M. Pauly. Realtime performance-based facial animation. *ACM Trans. Graph.*, 2011.

[22] T. Weise, B. Leibe, and L. V. Gool. Accurate and robust registration for in-hand modeling. CVPR, 2008.

[23] T. Weise, T. Wismer, B. Leibe, and L. Van Gool. In-hand scanning with online loop closure. 3DIM, 2009.