IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, May 2014.

Efficient Deformable Registration of Multi-Resolution Surfel Maps for Object Manipulation Skill Transfer

Jörg Stückler and Sven Behnke

***Abstract*— Endowing mobile manipulation robots with skills to use objects and tools often involves the programming or training on specific object instances. To apply this knowledge to novel instances from the same class of objects, a robot requires generalization capabilities for control as well as perception. In this paper, we propose an efficient approach to deformable registration of RGB-D images that enables robots to transfer skills between object instances. Our method provides a dense deformation field between the current image and an object model which allows for estimating local rigid transformations on the object's surface. Since we define grasp and motion strategies as poses and trajectories with respect to the object models, these strategies can be transferred to novel instances through local transformations derived from the deformation field. In experiments, we demonstrate the accuracy and run-time efficiency of our registration method. We also report on the use of our skill transfer approach in a public demonstration.**

I. INTRODUCTION

Devising manipulation control and perception capabilities for robots that generalize well to novel objects and tools is a challenging task. In this paper, we mainly focus on the perception part. Objects with the same function frequently share a common topology of functional parts such as handles and tool-tips. In this case, shape correspondences can be interpreted to also establish correspondences between the functional parts. In many object manipulation scenarios, controllers can be specified for specific object instances through grasp poses and 6-DoF trajectories relative to the functional parts. One can pose the problem of skill transfer as establishing correspondences between the object shapes, i.e., between the functional parts. Grasps and motions are then transferrable to novel object instances according to the shape deformation.

We propose an efficient deformable registration method that provides a dense displacement field between object shapes observed in RGB-D images. From the displacements, local transformations can be estimated between points on the object surfaces. We apply these local transformations to transfer grasps and motion trajectories between the objects.

Our registration approach is based on the coherent point drift (CPD) [1] algorithm. We extend it through efficient coarse-to-fine registration of RGB-D measurements. Instead of processing the raw pixels of the images, we represent the images in multi-resolution surfel maps (MRSMaps) [2], a compact 3D multi-resolution representation that stores the joint shape and color statistics contained in the image within an octree. In experiments, we demonstrate the accuracy and run-time efficiency of our registration method, being superior to plain processing of RGB-D images. We also report on the public demonstration of our approach as a key component for object manipulation skill transfer.

All authors are with Autonomous Intelligent Systems Group, Computer Science Institute VI, University of Bonn, 53113 Bonn, Germany stueckler@ais.uni-bonn.de, behnke@cs.uni-bonn.de

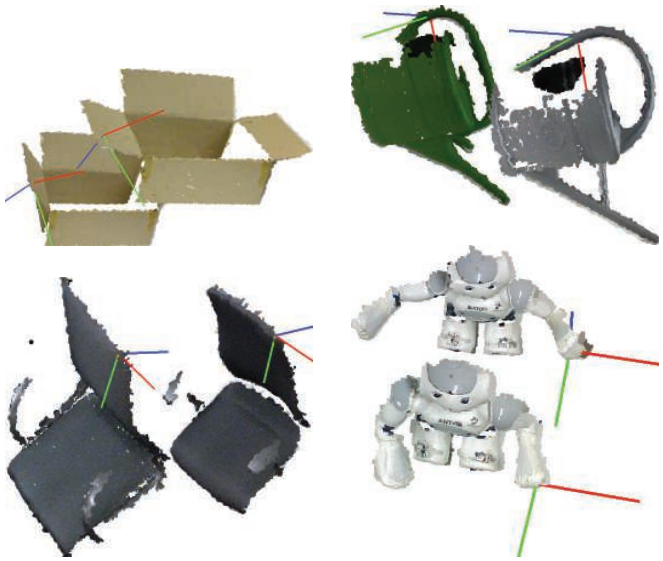

Fig. 1. We estimate local transformations between objects using deformable registration. This allows to transfer grasp poses and motion trajectories defined on local reference frames (e.g., handles or tool-tips) on model objects to novel object instances.

II. RELATED WORK

Many approaches to deformable registration represent scene and model surface by meshes or point clouds and estimate the local deformation of vertices or points. For example, Allen et al. [3] learn a shape-space of human bodies through deformable registration. They adapt the iterative closest points (ICP) algorithm to perform deformable registration between measured meshes of persons. Instead of estimating a single global rigid transformation, they determine a local rigid transformation at each vertex through energy minimization. The data terms of the energy capture the squared distance of vertices towards the closest counterparts in the other mesh after the transformation has been applied. To enforce smoothness of local transformations of neighboring vertices in the mesh, the difference between the local transforms is minimized concurrently. Amberg et al. [4] take a similar approach to align arbitrary meshes. They, however, allow for local affine transformations at the

vertices. In addition to local transforms at each vertex, Li et al. [5] include a global rigid transformation that acts on the complete mesh. Their energy formulation facilitates rigidness of the local affine transformations. The approach of Willimon et al. [6] enforces alignment of boundaries to register RGB-D images of clothing.

The above methods establish only a single correspondence for each point or vertex. It has been observed that both the basin of convergence and the accuracy can be improved by allowing each surface element to be softly assigned with multiple elements of the other surface. Anguelov et al. [7] model the correspondence of vertices between scene and model in a Markov random field (MRF) and infer the maximum likelihood (ML) correspondences through loopy belief propagation. The unary potentials measure the similarity in spin image descriptors [8], while pairwise potentials prefer to keep discrete nearness and farness relations. Myronenko and Song [1] and Jain and Vemuri [9] model the point clouds in Gaussian mixture models (GMMs). The CPD method [1] estimates probabilistic assignments of points and optimizes for the displacement field between source and model. Spatial smoothness of the solution is obtained through regularizing higher-order derivatives in the displacement field using a Gaussian kernel. Jain and Vemuri [9] impose GMMs on both point sets and minimize the L_2-norm between the mixture densities. Sagawa et al. [10] extend the non-rigid ICP method in [3] with soft assignments. Very recently, [11] also proposed an approach based on non-rigid registration in which motion trajectories are transfered between shape variants of objects. They use thin-plate splines to regularize the displacement field.

In the context of stereo and depth image processing, scene flow methods also recover displacement fields. For instance, the approach by Herbst et al. [12] computes 3D flow in RGB-D image pixels in a regularized variational framework. It requires about 8 to 30 seconds on a CPU for processing a 320×240 image.

Most of the presented methods focus on best accuracy but often neglect run-time efficiency. In this work, we develop an efficient deformable registration method based on CPD that aligns RGB-D images efficiently while being sufficiently accurate for robotic applications. To gain efficiency, we transform the RGB-D images into MRSMaps and match surfels from coarse to fine resolutions. Our approach seamlessly integrates color and contour cues with shape alignment to guide the soft assignments between the images and to improve accuracy. If a model is given a-priori, significant computational load can be transferred to pre-processing that only needs to be done once for the model. Our method then aligns images at a rate of 1 to 5 Hz on a CPU.

III. COHERENT POINT DRIFT

The CPD method [1] performs deformable registration between two point clouds: We denote $X = (x_1, \ldots, x_N)^T$ as the scene and $Y = (y_1, \ldots, y_M)^T$ as the model point cloud with D-dimensional points $x_i, y_j \in \mathbb{R}^D$. We assume that the surface underlying the model point cloud has been deformed towards the scene surface according to the displacement field $v : \mathbb{R}^D \to \mathbb{R}^D$ such that points y_j in the model cloud transform to a point $y_j + v(y_j)$ on the scene surface. The aim of the CPD method is to recover this displacement field.

A. Mixture Model for Observations

CPD explains the scene point cloud X as a set of samples from a mixture model on the deformed model cloud Y,

$$p(x_i \mid v, \sigma) = \sum_{j=1}^{M+1} p(c_{i,j})\ p(x_j \mid c_{i,j}, v, \sigma), \tag{1}$$

where $c_{i,j}$ is a shorthand for the 1-of-$(M+1)$ encoding binary variable $c_i \in \mathbb{B}^{M+1}$ with j-th entry set to 1. Naturally, c_i indicates the association of x_i to exactly one of the mixture components. The model is a Gaussian mixture on the M deformed model points and an additional uniform component,

$$p(x_i \mid v, \sigma) = \sum_{j=1}^{M} p(c_{i,j})\ \mathcal{N}(x_i; y_j + v(y_j), \sigma^2) + p(c_{i,M+1})\ p(x_i \mid c_{i,M+1}), \tag{2}$$

where σ is a standard deviation which is shared across all Gaussian mixture components. The uniform component generates each sample in X with equal probability $p(x_i \mid c_{i,M+1}) = \frac{1}{N}$. Its prior probability $w := p(c_{i,M+1})$ is a parameter that is chosen according to the noise inherent to the data. If we further assume equal prior likelihood for the association to each Gaussian mixture component, we obtain $p(c_{i,j}) = (1-w)\frac{1}{M}$ for all $j \in \{1, \ldots, M\}$. By modeling the scene points as samples from a mixture model on the model cloud, the CPD method does not make a hard association decision between the point sets, but a scene point is associated to every model point. The probability $p(c_{i,j} \mid x_i, v, \sigma)$ quantifies the likelihood of the assignment of x_i to the model point y_j.

B. Registration through Expectation-Maximization

The displacement field v is estimated through maximization of the logarithm of the joint data-likelihood

$$\ln p(X \mid v, \sigma) = \sum_{i=1}^{N} \ln \sum_{j=1}^{M+1} p(c_{i,j})\ p(x_i \mid c_{i,j}, v, \sigma). \tag{3}$$

While a direct optimization of this objective function is not feasible, it lends itself to the EM method [13]. The component associations $c = \{c_1, \ldots, c_N\}$ are treated as the latent variables to yield the EM objective

$$L(q, v, \sigma) := \sum_{i=1}^{N} \sum_{j=1}^{M+1} q(c_{i,j}) \ln \frac{p(c_{i,j})\ p(x_i \mid c_{i,j}, v, \sigma)}{q(c_{i,j})}, \tag{4}$$

by exploiting $q(c) = \prod_{i=1}^{N} \prod_{j=1}^{M+1} q(c_{i,j})$. In the M-step, the latest estimate $\overline{q}$ for the distribution over component associations is held fixed to optimize for the displacement field v and standard deviation σ

$$\{\widehat{v}, \widehat{\sigma}\} = \arg\max_{v,\sigma}\ L(\overline{q}, v, \sigma) \tag{5}$$

with

$$L(\overline{q}, v, \sigma) := \sum_{i=1}^{N} \sum_{j=1}^{M+1} \overline{q}(c_{i,j}) \ln p(c_{i,j})\ p(x_i \mid c_{i,j}, v, \sigma) \tag{6}$$

$$= const. - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} \overline{q}(c_{i,j}) \cdot \tag{7}$$

$$\left(D \ln(2\pi\sigma^2) + \frac{1}{\sigma^2} \left\| x_i - (y_j + v(y_j)) \right\|_2^2 \right). \tag{8}$$

The E-step obtains a new optimum $\widehat{q}$ for the distribution q by the conditional likelihood of the cluster associations given the latest displacement field estimate $\overline{v}$ and standard deviation $\overline{\sigma}$

$$\widehat{q}(c_{i,j}) = \frac{p(c_{i,j})\ p(x_i \mid c_{i,j}, \overline{v}, \overline{\sigma})}{\sum_{j'=1}^{M} p(c_{i,j'})\ p(x_i \mid c_{i,j'}, \overline{v}, \overline{\sigma})}. \tag{9}$$

For the Gaussian mixture components this corresponds to

$$\widehat{q}(c_{i,j}) = \frac{\exp\left(-\frac{1}{2\overline{\sigma}^2} \left\| x_i - (y_j + v(y_j)) \right\|_2^2 \right)}{\gamma + \sum_{j'=1}^{M} \exp\left(-\frac{1}{2\overline{\sigma}^2} \left\| x_i - (y_j + v(y_j)) \right\|_2^2 \right)}. \tag{10}$$

with $\gamma := (2\pi\sigma^2)^{D/2} \frac{w}{1-w} \frac{M}{N}$.

C. Regularized Deformation Field

It is a well known fact that estimating a function with many degrees of freedom from a set of samples purely from the data-likelihood easily is an ill-posed problem [14]. Myronenko and Song [1] augment the joint data-likelihood in Eq. (3)

$$\ln p(X, v \mid \sigma) = \ln p(X \mid \sigma, v) - \frac{\lambda}{2} \left\| v \right\|_{\mathcal{H}}^2. \tag{11}$$

with Tikhonov regularization [14] by choosing the norm in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$. It is straightforward to extend the EM approach of the previous Sec. III-B to the joint likelihood of data and displacement field:

$$L^{regularized}(q, v, \sigma) := \ln p(v) + \sum_{i=1}^{N} \sum_{j=1}^{M+1} q(c_{i,j}) \ln \frac{p(c_{i,j})\ p(x_i \mid c_{i,j}, v, \sigma)}{q(c_{i,j})}. \tag{12}$$

Myronenko and Song [1] apply a Gaussian reproducing kernel $g(y, y') := \exp\left(-\frac{\left\| y - y' \right\|_2^2}{2\beta^2} \right)$ to penalize high frequencies in the displacement field. A norm $\left\| Pv \right\|^2$ on the outcome of a linear differential operator P applied to v also induces a RKHS [15]. The reproducing kernel $k(y, y')$ is equivalent to the Green's function of the differential operator P^*P, where P^* is the adjoint operator to P. The kernel hence defines a right-inverse integral operator to the differential operator P^*P. Conversely, we can find a linear differential operator P for any RKHS [15], [16].

D. Regularized Maximization Step

In the M-step, we optimize (12) for the displacement field v and the standard deviation σ. Since a joint closed-form solution is not available, we optimize for v and σ alternately.

1) Standard Deviation: Setting the derivative of Eq. (12) for the standard deviation σ to zero yields

$$\widehat{\sigma}^2 = \frac{1}{N_P D} \sum_{i=1}^{N} \sum_{j=1}^{M} \left\| x_i - (y_j + v(y_j)) \right\|_2^2, \tag{13}$$

where we define $N_P := \sum_{i=1}^{N} \sum_{j=1}^{M} \overline{q}(c_{i,j})$.

2) Deformation Field: Analogous to the derivation in [16], the Euler-Lagrange equation for the functional in Eq. (12) is obtained:

$$P^*P\, \widehat{v}(y) = \frac{1}{\sigma^2 \lambda} \sum_{i=1}^{N} \sum_{j=1}^{M} \overline{q}(c_{i,j})\ (x_i - (y_j + \widehat{v}(y_j)))\ \delta(y - y_j). \tag{14}$$

This partial differential equation can be solved using the Green's function $k(y, y')$ of the operator P^*P

$$\widehat{v}(y) = \int k(y, y') \frac{1}{\sigma^2 \lambda} \sum_{i=1}^{N} \sum_{j=1}^{M} \overline{q}(c_{i,j})\ (x_i - (y_j + \widehat{v}(y_j)))\ \delta(y' - y_j) dy' \tag{15}$$

such that

$$\widehat{v}(y) = \frac{1}{\sigma^2 \lambda} \sum_{i=1}^{N} \sum_{j=1}^{M} \overline{q}(c_{i,j})\ (x_i - (y_j + \widehat{v}(y_j)))\ k(y, y_j) \tag{16}$$

$$= \sum_{j=1}^{M} w_j k(y, y_j), \tag{17}$$

with weights $w_j := \frac{1}{\sigma^2 \lambda} \sum_{i=1}^{N} \overline{q}(c_{i,j})\ (x_i - (y_j + \widehat{v}(y_j)))$.

To obtain a solution, we need to evaluate $\widehat{v}(y)$ at the model points y_j and solve for the weights w_j. Let $W := (w_1, \ldots, w_M)^T \in \mathbb{R}^{M \times D}$ to write $v(y) = GW$ using the Gram matrix $G \in \mathbb{R}^{M \times M}$ with $G_{ij} := k(y_i, y_j)$. The weights for the solution $\widehat{v}(y)$ are

$$W = (dP1\, G + \lambda\sigma^2)^{-1} (PX - dP1\, Y), \tag{18}$$

where $P_{ji} := \overline{q}(c_{i,j})$ and $dP1 := \mathrm{diag}(P\, 1_{N \times 1})$ [1].

Note that the solution for the weights W in Eq. (18) requires the inversion of a potentially large $M \times M$ matrix whose size depends on the size of the model point cloud. To reduce complexity, Myronenko and Song [1] propose to utilize a low-rank approximation of G, $\widehat{G} := Q \Lambda Q^T$ with the matrix Q of eigenvectors and the diagonal matrix Λ containing the K largest eigenvalues of G. Using the Woodbury identity, Eq. (18) is reformulated to arrive at

$$W \approx \frac{1}{\lambda\sigma^2} \left(I - dP1\, Q \left(\lambda\sigma^2 \Lambda^{-1} + Q^T dP1\, Q \right)^{-1} Q^T \right) (PX - dP1\, Y). \tag{19}$$

The outer inversion acts on a $K \times K$ matrix, such that we can drastically improve run-time over the $M \times M$ matrix inversion

in Eq. (18) by choosing $K \ll M$. The low-rank approximation constrains the solution for the displacement field in a low-dimensional embedding, which further regularizes the displacement field.

IV. EFFICIENT DEFORMABLE REGISTRATION OF MULTI-RESOLUTION SURFEL MAPS

We propose a multi-resolution extension to the CPD method for efficient deformable registration of RGB-D images. Instead of processing the dense point clouds of the RGB-D images directly, we utilize multi-resolution surfel maps (MRSMaps) [2] to perform deformable registration on a compressed image representation. This image representation stores the joint color and shape statistics of points within 3D voxels (coined surfels) at multiple resolutions in an octree. The maximum resolution at a point is limited proportional to its squared distance in order to capture the error properties of the RGB-D camera. In effect, the map exhibits a local multi-resolution structure which well reflects the accuracy of the measurements and compresses the image from 640×480 pixels into only a few thousand surfels. [1]

We further improve the performance of the algorithm by aligning maps from coarse to fine resolutions. The registration on finer resolutions is initialized from the result on the coarser one. In addition to depth, we also utilize cues such as color and contours. We improve robustness and efficiency of our algorithm by using a modified Gaussian kernel with compact support.

A. Coarse-To-Fine Deformable Registration

The run-time complexity of the CPD algorithm depends at least quadratically on the size of the two point sets. If we do not apply the low-rank approximation, it is even cubic in the size of the model cloud—due to the inversion of the Gram matrix. By processing the resolutions from coarse to fine, we can keep the size of the point clouds as small as possible. The displacement field of coarse resolutions can be used to initialize the displacement on the next finer resolution such that the number of iterations required to converge is greatly decreased.

We represent both images by a scene and model MRSMap. The means of the surfels within each resolution $\rho(d)$ at depth d of the maps define scene and model point clouds $X_d := (x_{d,1}, \ldots, x_{d,N_d})$ and $Y_d := (y_{d,1}, \ldots, y_{d,M_d})$.

We iterate from coarse to fine resolutions, starting at the coarsest resolution $\rho(0)$ at depth 0 in the map. Let d be the current depth processed. Our aim is to find the displacement field v_d from scene to model point clouds X_d, Y_d and the standard deviation σ_d.

1) Per-Resolution Initialization: When transiting to the next finer resolution, the standard deviation $\sigma_d \leftarrow \sigma_{d-1}$ is initialized from the result σ_{d-1} of the previous iteration.

[1]Our MRSMap implementation is available open-source from http://code.google.com/p/mrsmap/ .

2) Full-Rank Optimization: We initialize the registration on each depth with the displacement field v_{d-1} of the previous coarser resolution. Each mean $y_{d,i}$ on the current depth is mapped to its displacement

$$v_{d-1}(y_{d,i}) = \sum_{j=1}^{M_{d-1}} w_{d-1,j}\ k(y_{d,i}, y_{d-1,j}) \tag{20}$$

according to the coarser resolution displacement field which we abbreviate as

$$v_{d-1}(Y_d) = G(Y_d, Y_{d-1})\ W_{d-1}, \tag{21}$$

where $G(Y_d, Y_{d-1}) \in \mathbb{R}^{M_d \times M_{d-1}}$ is a Gram matrix with $g_{ij} := k(y_{d,i}, y_{d-1,j})$. Subsequently, we utilize $v(Y_d) = G_d W_d$ to solve for the initial weight matrix

$$W_d \leftarrow G_d^{-1}\ G(Y_d, Y_{d-1})\ W_{d-1} \tag{22}$$

on the current depth.

3) Low-Rank Approximation: We compensate for the effect of the low-rank approximation on the found weights through

$$W_d \leftarrow \widehat{G}_d^{-1}\ G(Y_d, Y_{d-1})\ G_{d-1}^{-1}\ \widehat{G}_{d-1} W_{d-1}. \tag{23}$$

This approach requires the inversion of the low-rank approximation $\widehat{G}_d$ and the full-rank Gram matrix G_{d-1}^{-1}. While the former is in $\mathcal{O}(K^3)$ due to $\widehat{G}_d^{-1} = Q\Lambda^{-1}Q^T$, the latter is in $\mathcal{O}(M^3)$. Notably, both inversions could be precomputed once, for instance, if the model cloud is an object map, or for sequential registration of scene maps towards a persisting model map. For the inversion of the Gram matrix, it must be well-conditioned.

4) Resolution-Dependent Kernel with Compact Support: Gaussian kernels produce a dense Gram matrix with potentially very small entries. The smaller the scale β, the larger the condition number of the Gram matrix and, hence, the less numerically stable is the inversion of the Gram matrix [17]. Furthermore, sparse matrices can be inverted much more efficiently than dense matrices using sparse matrix factorizations such as the LU- or Cholesky-decompositions. We therefore use a modified Gaussian kernel with compact support [18] instead, i.e.,

$$k(y, y') = \varphi_{l,k}(y, y')\ g(y, y'), \tag{24}$$

where $\varphi_{l,k} \in \mathcal{C}^{2k}$ is a Wendland kernel [19] with $l = \lfloor D/2 \rfloor + k + 1 \in \mathbb{N}$. Due to our 7-dimensional points, we choose $\varphi_{5,1}(y, y')$.

We adapt the scale $\beta_d = \beta_0\, \rho(d)^{-1}$ of the kernel $k_d(y, y')$ to the current resolution $\rho(d)$. This way, spatial smoothing is performed from low to high frequencies which is required as high frequencies in the displacement field are only observable on fine resolutions due to the sampling theorem.

5) Handling of Resolution-Borders: Since we use a distance-dependent resolution limit in MRSMaps, surfels have redundant counterparts in ancestor nodes on coarser resolutions, but they may not be represented at finer resolutions. This leads to surfels whose local context is in parts only

present at coarser resolutions. We denote the set of surfels with this property as resolution border surfels.

We still constrain the deformation of resolution border surfels to the displacement field in the complete local context of the surfels. We include the means X_{d-1} of the scene surfels from the previous coarser resolution. Secondly, we add a prior on the displacement field v_d to Eq. (11),

$$\ln p(X_d, v_d \mid \sigma_d, v_{d-1}) = \ln p(X_d \mid \sigma_d, v_d) + \ln p(v_d \mid v_{d-1}) - \frac{\lambda}{2} \|v_d\|^2_{\mathcal{H}}, \quad (25)$$

to favor compatibility with the displacement field v_{d-1} of the coarser resolution at the resolution border surfels. We need to consider this prior in the M-step.

Let $\tilde{Y}_d \subseteq Y_d$ be the means of the resolution border surfels at the current resolution. We model the prior

$$\ln p(v_d \mid v_{d-1}) := -\frac{1}{2} \sum_{j=1}^{M_d} \gamma(y_{d,j}) \, \|v_d(y_{d,j}) - v_{d-1}(y_{d,j})\|_2^2, \quad (26)$$

$$\text{with } \gamma(y_{d,j}) := \begin{cases} \sigma_\gamma^{-2} & \text{if } y_{d,j} \in \tilde{Y}_d \\ 0 & \text{otherwise.} \end{cases} \quad (27)$$

We adapt $\sigma_\gamma := \sigma_{\gamma,0} \, \rho(d)^{-1}$ to the current resolution.

With this additional prior term, we obtain

$$P^* P \; \widehat{v}_d(y) = \frac{1}{\sigma_d^2 \lambda} \sum_{j=1}^{M_d} w'_{d,j} \; \delta(y - y_j), \quad (28)$$

where we now define

$$w'_{d,j} := \frac{1}{\sigma_d^2 \lambda} \left(\sum_{i=1}^{N_d} \overline{q}(c_{i,j}) \; (x_{d,i} - (y_{d,j} + \widehat{v}_d(y_{d,j}))) \right) + \frac{1}{\lambda} \gamma(y_{d,j}) \, (v_{d-1}(y_{d,j}) - \widehat{v}_d(y_{d,j})) \, . \quad (29)$$

Using the Green's function $k(y, y')$, we solve for $\widehat{v}_d(y)$ and obtain the linear system of equations

$$\left(\sigma_d^2 \, \lambda \, I + \left(dP1 + \sigma_d^2 \, d\Gamma\right) G_d\right) \; W'_d = PX_d - dP1 \, Y_d + \sigma_d^2 \, d\Gamma \, v_{d-1}(Y_d), \quad (30)$$

where we use the shorthand $d\Gamma := \mathrm{diag}(\gamma(Y_d))$. It's low-rank approximation is

$$W'_d \approx \frac{1}{\lambda \sigma_d^2} \Big(I - \left(dP1 + \sigma_d^2 \, d\Gamma\right) \; Q_d \left(\lambda \sigma_d^2 \Lambda_d^{-1} + Q_d^T \left(dP1 + \sigma_d^2 \, d\Gamma\right) \; Q_d\right)^{-1} Q_d^T \Big) \left(PX_d - dP1 \, Y_d + \sigma_d^2 \, d\Gamma \, v_{d-1}(Y_d)\right) \quad (31)$$

with $\widehat{G}_d = Q_d \Lambda_d Q_d^T$.

B. Color and Contour Cues

The CPD method is not limited to registration in the spatial domain. We use the full six-dimensional spatial and color mean of the surfels. In addition, we add contours determined as surfels at foreground borders as a seventh point dimension. We set the contour value of a point to β_d if it is on a foreground border, or 0 otherwise. This places points closer in feature space that are either on or off contours.

C. Convergence Criterion

Our convergence criterion examines the relative change

$$\Delta L_t := \left| \frac{L_t - L_{t-1}}{L_{t-1}} \right|, L_t := \frac{1}{2} \lambda \, \|v_{d,t}\|^2_{\mathcal{H}} \quad (32)$$

in the norm of the displacement field

$$\|v_{d,t}\|^2_{\mathcal{H}} = \mathrm{tr}(W_{d,t}^T \, G_{d,t} \, W_{d,t}). \quad (33)$$

If this rate decreases below a threshold, the estimate of the displacement field is assumed to have converged.

V. LOCAL DEFORMATIONS

The continuous displacement field allows us to estimate the local infinitesimal deformation at any point in terms of translation and rotation between both surfaces. These local deformation quantities can be estimated in each direction between scene and model surface. Since the displacement field is defined to act on points on the model surface, we begin our investigation in the direction from model to scene.

A. Local Deformations from Model to Scene

1) Full-Rank Optimization: It is well known in continuum mechanics [20] how infinitesimal local deformations can be estimated from a continuous deformation function $\phi : \mathbb{R}^3 \mapsto \mathbb{R}^3$ that maps the position of infinitesimal particles in an elastic body to their deformed location. Our displacement field v defines such a deformation function in a straightforward way,

$$\phi(y) := y + v(y). \quad (34)$$

The infinitesimal deformation at a point y is then specified by the Jacobian of the deformation function at y,

$$\nabla_y \phi(y) = I + \nabla_y v(y). \quad (35)$$

As long as we use differentiable kernels in our estimation algorithm, we may write

$$\nabla_y \phi(y) = I + \sum_{i=1}^{M} w_i \nabla_y k(y_i, y). \quad (36)$$

Rotation $R(y)$ and strain $S(y)$ are obtained through polar decomposition of the Jacobian $\nabla_y \phi(y) = RU$, i.e., $R(y) = UV^T$ and $S(y) = V \Sigma V^T$, where $\nabla_y \phi(y) = U \Sigma V^T$ is the singular value decomposition of the Jacobian. The translation $t(y) = v(y)$ is set to the displacement at y.

To query the local deformation of a point y from a deformable registration result for MRSMaps, we first find the finest resolution $\rho(d)$ in which the point y is represented in the model map. Translation, rotation, and strain are then determined via the displacement field v_d.

2) Low-Rank Approximation: If we use a low-rank-approximation, the weights W of the displacement field v are computed with respect to a low-dimensional embedding of the kernel $k(y, y')$. Hence, Eq. (36) is not directly applicable. Instead we estimate translation and rotation from the local displacements around y using the method in [21]. We locally weigh neighboring displacements with a Gaussian window function.

B. Local Deformations from Scene to Model

1) Full-Rank Optimization: A closed-form solution to the local deformations from scene to model would require the inverse $v^{-1}(x)$ of the displacement field v for a scene point x. Since such an inverse is not available, we approximate the inverse displacement

$$v^{-1}(x) = -\frac{\sum_{i=1}^{M} g(x, y_i + v(y_i), r) v(y_i)}{\sum_{i=1}^{M} g(x, y_i + v(y_i), r)}. \quad (37)$$

with the displacements of model points y_i that deform close to x. We can then use the closed-form approach in Sec. V-A.1 to determine the local rotation $R(x) = R(x+v^{-1}(x))^T$. The translation is $t(x) = v^{-1}(x)$.

2) Low-Rank Approximation: For estimating rotation and translation while using low-rank-approximations, we determine rotation and translation from displacements local to the queried scene point as in Sec. V-A.

VI. TRANSFER OF OBJECT MANIPULATION SKILLS

We apply our deformable registration method for object manipulation skill transfer. Once pre-grasp and grasp poses are defined for an object instance, these grasps are transferred to other instances of the same object class. Similarly, motion controllers that move a reference frame on the object can be adapted to different shapes within the object class.

In our approach, we first segment the object of interest in the RGB-D image using techniques such as support-plane segmentation [22]. The RGB-D image segment is then transformed into a MRSMap and a reference object model MRSMap is aligned with the image. The grasp poses and motion trajectories are defined in terms of local coordinate frames relative to the object's reference frame. We assume that the poses and trajectories are close to the reference object's surface, and, hence, we find the local rigid transformation from the reference object towards the image segment using one of the methods in Sec. V. Finally, the motions are executed according to the transformed grasp and motion trajectories.

VII. RESULTS

A. Quantitative Evaluation

We evaluate accuracy and run-time of our registration approach on synthetically deformed RGB-D images. For our experiments, we used an Intel Core i7-4770K CPU (max. 3.50 GHz) and 32 GB of RAM and chose two sequences of the RGB-D benchmark dataset [23]. In the freiburg2_desk sequence, the camera observes a table-top scene. The planar surfaces create local aperture problems that need to be adressed by smoothness regularization. The freiburg3_teddy sequence contains views on a teddy bear with salient yellow and brown coloring. We process 500 frames per sequence to assess the accuracy of our method in recovering displacements as well as the run-time required to align the images.

We synthetically generate deformations in order to have ground truth available for assessing registration accuracy. Each frame is randomly disturbed by adding Gaussian noise to the 3D Euclidean dimensions. We sample the Gaussian noise in image coordinates and choose a standard deviation uniformly between 100 and 200 pixels in the x- and y- direction of the image separately. Each of ten Gaussians applies up to 0.1 m distortion. In total, we normalize the applied deformation to a maximum of 0.1 m in each direction.

We assess the performance of several variants of our approach. Full-rank methods are marked by "F", whereas we denote low-rank approximations by "L". The variants F– and F-+ do not use color for registration, while the second sign indicates the use of the contour cue. The methods tagged with "*" do not include surfels from coarser resolutions from the scene cloud and do not constrain the displacement field on the resulting field of the coarser resolution (but we initialize it from the coarser resolutions and perform coarse-to-fine registration). For all full-rank approaches, we set $\beta_0 = 160$. The low-rank approximations have been run with $\beta_0 = 20$.

Tables I and II summarize the average run-time in milliseconds spent per frame. Using additional cues such as color and contours increases the run-time slightly. The variants utilizing low-rank approximations are significantly faster in the registration step, while the preparation step is more expensive. We note that this preparation step would only be needed to be executed once for a fixed object model. In this case, our low-rank coarse-to-fine registration method achieves a frame rate between 1 to 5 Hz. Note that the run-time of plain concurrent processing of all the surfels in the MRSMap requires run-time of 10 to 30 seconds per image using low-rank approximations.

We also compared our approach to plain registration of RGB-D images using the CPD approach. For a fair comparison, we project synthetically deformed RGB-D point clouds back into RGB-D images and process the images with our multi-resolution approach as well as with plain CPD registration. Due to memory limitations, the plain registration method could only process images at a downsampling factor of 8 (resolution 80×60), while our approach integrates full VGA (640×480) resolution images in MRSMaps. While with low-rank approximations plain registration requires 4.74 s in average on 200 images of the freiburg2_desk sequence, our approach only takes 1.29 s.

Figs. 2 and 3 demonstrate the accuracy of our approach. Using color and contour cues gives best performance on the finest resolution (0.025 m). Not using color, contours, or coarse-to-fine registration degrades performance. We also notice that using a low-rank approximation is only slighly less accurate than the full-rank methods. Our coarse-to-fine method also performs more accurately compared to plain registration. In the mean, it achieves a deviation of 0.0178 m from the ground-truth displacements (mean 0.0755 m). Plain registration yields 0.0482 m mean deviation for average ground-truth displacements of 0.0752 m. While we used color and contours for both methods and the same parameters, we set the scale of the smoothing kernel equivalent to the scale for the finest resolution used in our MRSMap approach. Our multi-resolution approach seems to handle the

TABLE I

COMPARISON OF AVERAGE RUN-TIME IN MILLISECONDS PER IMAGE USING FULL-RANK GRAM MATRICES.

sequence	F	F-+	F+-	F–	F*
fr2 desk, prepare	344	330	340	**325**	344
fr2 desk, register	7802	6621	**1826**	1848	5278
fr2 desk, total	8216	7020	**2235**	2243	5693
fr3 teddy, prepare	141	135	139	**133**	141
fr3 teddy, register	3697	3340	**1367**	1494	3435
fr3 teddy, total	3921	3559	**1589**	1711	3659

TABLE II

COMPARISON OF AVERAGE RUN-TIME IN MILLISECONDS PER IMAGE USING THE LOW-RANK APPROXIMATION TO THE GRAM MATRIX.

sequence	L	L-+	L+-	L–	L*
fr2 desk, prepare	437	423	433	**417**	438
fr2 desk, register	643	464	553	**348**	425
fr2 desk, total	1149	957	1056	**835**	933
fr3 teddy, prepare	222	216	220	**214**	221
fr3 teddy, register	467	335	390	**268**	290
fr3 teddy, total	772	634	693	**565**	594

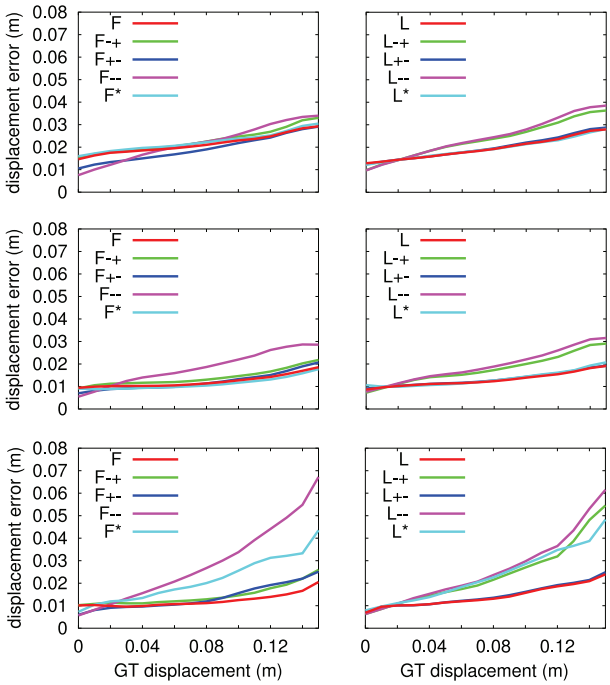

Fig. 2. Median accuracy for deformable registration of synthetically deformed RGB-D images on the freiburg2_desk dataset. Top: 0.1 m, middle: 0.05 m, bottom: 0.025 m resolutions.

varying Euclidean sampling rate in the image better.

B. Non-Rigid Registration and Local Deformation Examples

In Fig. 4, we show typical results of our low-rank deformable registration method on RGB-D image segments of objects. Examples for estimated local transformations can be found in Fig. 1. The local coordinate frames are well displaced to their counterparts in both image segments. Also the orientation reflects the local bending of the surface.

C. Public Demonstration of Manipulation Skill Transfer

We publicly demonstrated our deformable registration approach in a mobile manipulation scenario during the Open Challenge at RoboCup 2013 in Eindhoven, Netherlands[2]. Our robot Cosero transferred watering can manipulation skills to a novel can. Fig. 5 shows images taken during the demonstration. A short video clip accompanies this paper. The demonstration was well received by the jury consisting of team leaders and received high scores. Overall, we won the 2013 RoboCup@Home competition.

VIII. CONCLUSIONS

In this paper we proposed an efficient variant of the coherent point drift (CPD) algorithm for deformable registration of RGB-D images. Our approach performs coarse-to-fine alignment of surfels at multiple 3D resolutions and estimates a displacement field on every resolution.

We evaluated the run-time and accuracy of our method on a synthetically generated dataset with ground truth displacement information. Our approach yields good accuracy and low run-times. For registering object models, our method achieves a frame rate of 1 to 5 Hz on a CPU.

We develop the method for object manipulation skill transfer. Many skills can be represented as a set of grasp and motion trajectories relative to the local reference frame of the object. From the displacement field provided by our registration method, we can estimate the local transformation of such grasps and motions. We demonstrated this procedure publicly for transferring a bimanual grasp from one watering can to another. The approach has also been used to perform the watering motion, in which the motion of the can end-effector has been predefined for the original object model.

In future work, we will consider parallel implementations on GPU to facilitate real-time deformable registration. The accuracy and basin of convergence could possibly be further improved by integrating higher-order features.

[2]video available from http://www.youtube.com/watch?v=I1kN1bAeeB0

REFERENCES

[1] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE Trans. on PAMI*, vol. 32, no. 12, pp. 2262–2275, 2010.

[2] J. Stückler and S. Behnke, "Multi-resolution surfel maps for efficient dense 3D modeling and tracking," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 137–147, 2014.

[3] B. Allen, B. Curless, and Z. Popović, "The space of human body shapes: reconstruction and parameterization from range scans," *ACM Transations on Graphics*, vol. 22, no. 3, pp. 587–594, July 2003.

[4] B. Amberg, S. Romdhani, and T. Vetter, "Optimal step nonrigid icp algorithms for surface registration," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.

[5] H. Li, R. W. Sumner, and M. Pauly, "Global correspondence optimization for non-rigid registration of depth scans," *Computer Graphics Forum (Proc. SGP'08)*, vol. 27, no. 5, July 2008.

[6] B. Willimon, I. Walker, and S. Birchfield, "3D non-rigid deformable surface estimation without feature correspondence," in *Proc. of the IEEE Int. Conference on Robotics and Automation (ICRA)*, 2013.

Fig. 5. Cognitive service robot Cosero manipulates a novel watering can during the Open Challenge at RoboCup 2013 in Eindhoven, Netherlands. We specified bimanual grasp poses, the can's end-effector, and the motion of the end-effector for watering a plant for another watering can instance. Cosero used our deformable registration method to efficiently align the can in its current RGB-D image with the model can. From the displacement field Cosero estimates the poses of the grasps and the watering can's end-effector using our proposed local transformation estimation method. It then grasps the watering can and waters a plant using the generalized skill.

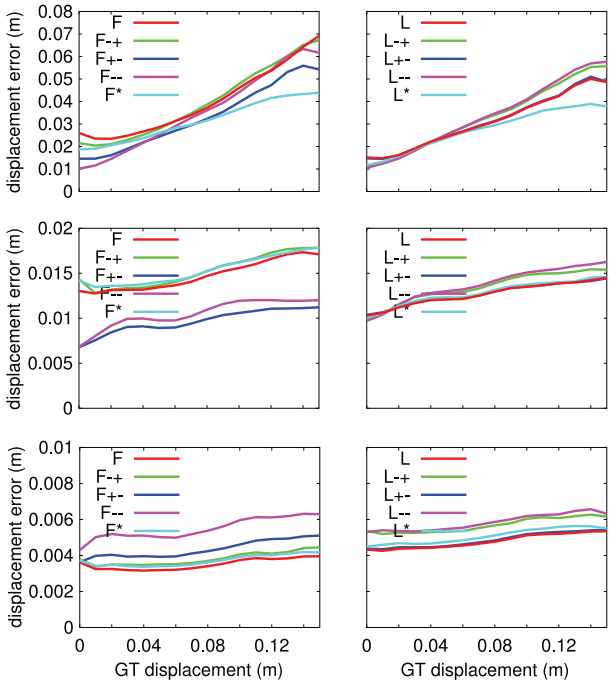

Fig. 3. Median accuracy for deformable registration of synthetically deformed RGB-D images on the freiburg3_teddy dataset. Top: 0.1 m, middle: 0.05 m, bottom: 0.025 m resolutions.

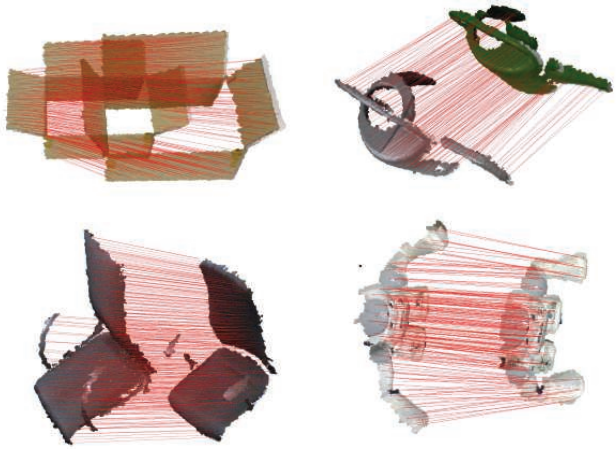

Fig. 4. Deformable registration examples.

[7] D. Anguelov, P. Srinivasan, H.-C. Pang, D. Koller, S. Thrun, and J. Davis, "The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces." in *Proc. of the International Conference on Advances in Neural Information Processing (NIPS)*, 2004.

[8] A. Johnson, "Spin-images: A representation for 3-D surface matching," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 1997.

[9] B. Jian and B. C. Vemuri, "Robust point set registration using Gaussian mixture models," *IEEE Transations on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1633–1645, 2011.

[10] R. Sagawa, K. Akasaka, Y. Yagi, H. Hamer, and L. Van Gool, "Elastic convolved ICP for the registration of deformable objects," in *Proceedings of the IEEE Int. Conf. on Computer Vision Workshops (ICCV Workshops)*, 2009, pp. 1558–1565.

[11] J. Schulman, A. Gupta, S. Venkatesan, M. Tayson-Frederick, and P. Abbeel, "A case study of trajectory transfer through non-rigid registration for a simplified suturing scenario," in *Proc. of the 26th IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[12] E. Herbst, X. Ren, and D. Fox, "RGB-D flow: Dense 3-D motion estimation using color and depth," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., 2006.

[14] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. John Wiley & Sons, New York,, 1977.

[15] A. J. Smola, B. Schölkopf, and K.-R. Müller, "The connection between regularization operators and support vector kernels," *Neural Networks*, vol. 11, no. 4, pp. 637–649, June 1998.

[16] Z. Chen and S. Haykin, "On different facets of regularization theory." *Neural Computation*, vol. 14, no. 12, pp. 2791–2846, 2002.

[17] B. Fornberg and J. Zuev, "The runge phenomenon and spatially variable shape parameters in RBF interpolation," *Computers & Mathematics with Applications*, vol. 54, no. 3, pp. 379 – 398, 2007.

[18] M. G. Genton, "Classes of kernels for machine learning: a statistics perspective," *J. Mach. Learn. Res.*, vol. 2, pp. 299–312, Mar. 2002.

[19] H. Wendland, "Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree," *Advances in Computational Mathematics*, vol. 4, no. 1, pp. 389–396, 1995.

[20] R. Batra, *Elements of Continuum Mechanics*, ser. AIAA education series. American Institute of Aeronautics and Astronautics, 2006.

[21] K. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 5, pp. 698–700, 1987.

[22] J. Stückler, R. Steffens, D. Holz, and S. Behnke, "Real-Time 3D Perception and Efficient Grasp Planning for Everyday Manipulation Tasks," in *Proceedings of the European Conference on Mobile Robots (ECMR)*, Örebro, Sweden, September 2011, pp. 177–182.

[23] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. of the Int. Conference on Intelligent Robot Systems (IROS)*, 2012.