Visual Articulated Tracking in the Presence of Occlusions

Christian Rauch[1], Timothy Hospedales[1], Jamie Shotton[2], Maurice Fallon[3]

***Abstract*— This paper focuses on visual tracking of a robotic manipulator during manipulation. In this situation, tracking is prone to failure when visual distractions are created by the object being manipulated and the clutter in the environment. Current state-of-the-art approaches, which typically rely on model-fitting using Iterative Closest Point (ICP), fail in the presence of distracting data points and are unable to recover. Meanwhile, discriminative methods which are trained only to distinguish parts of the tracked object can also fail in these scenarios as data points from the occlusions are incorrectly classified as being from the manipulator. We instead propose to use the per-pixel data-to-model associations provided from a random forest to avoid local minima during model fitting. By training the random forest with artificial occlusions we can achieve increased robustness to occlusion and clutter in the scene. We do this without specific knowledge about the type or location of the manipulated object. Our approach is demonstrated by using dense depth data from an RGB-D camera to track a robotic manipulator during manipulation and in presence of occlusions.**

I. Introduction

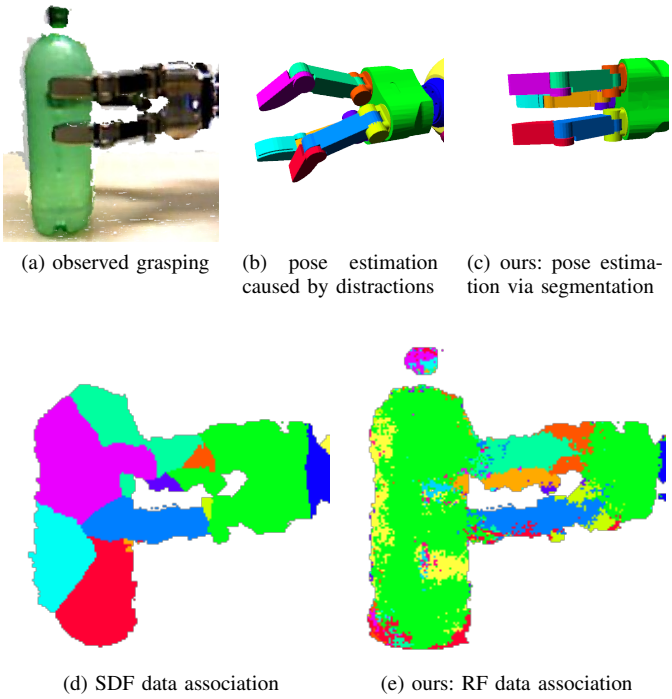
When estimating the state of a robot during manipulation, the common approach is to use joint sensing, forward kinematics (FK) and a complete description of the kinematic model of the robot to compute the position of the end effector. However joint sensing can be affected by calibration inaccuracies, quantisation noise and non-linearities, or may not be available at all for underactuated and dexterous manipulators. This traditional industrial approach does not consider tactile information nor does it incorporate visual sensing, which is of course heavily used during human manipulation. Finally, it cannot track the state of the manipulated object.

We are motivated to explore combined visual tracking of manipulator and object by the prior work of [1], [2]. A key challenge for visual tracking is the presence of distractor objects. These objects occlude the manipulator and add irrelevant visual information which can dis-improve tracking.

Estimating the full and valid configuration of an articulated object directly from images is a challenging problem. In this work we propose similar to [3], [4], [5], [6] to combine model-based tracking, which simplifies the kinematically plausible state estimation, with discriminative information to prevent failures due to the distracting visual information.

The core contribution of this paper is the integration of pixel-wise predictions from a random forest into a model-fitting framework that is robust to incorrect initialisation and un-modelled occlusions, as illustrated in Figure 1.

[1]Institute of Perception, Action and Behaviour, School of Informatics, University of Edinburgh, UK Christian.Rauch@ed.ac.uk
[2]Microsoft, Cambridge, UK
[3]Oxford Robotics Institute, Department of Engineering Science, University of Oxford, UK mfallon@ox.ac.uk



(a) observed grasping (b) pose estimation caused by distractions (c) ours: pose estimation via segmentation

(d) SDF data association (e) ours: RF data association

Fig. 1: Tracking the pose of the manipulator (coloured mesh) during a grasping task (a). The baseline approach (b) assigns data points of the manipulated object to finger parts (d) resulting in a shift of the palm towards the bottle. Our approach (c) keeps the palm pose estimate stabilised by the correct classification of palm and fingers, despite the distracting pixels of the manipulated object being incorrectly classified (e).

II. Related Work

We categorise visual articulated tracking into **generative model-fitting** and **discriminative** approaches as well as **hybrid** methods which combine generative model-fitting with discriminative information. In the following we give a brief overview of the relevant literature for each approach.

Generative model-fitting: Given a model of the tracked object, generative model fitting aims to synthesise a set of hypotheses of the model's state and compare these hypotheses with the observed state. These methods rely on a good metric to quantify the similarity between the synthesised state and the real observation (the objective function), and an efficient method for exploring the large state space of the articulated model.

Early work by Oikonomidis *et al.* [1] used colour and edge

Preprint submitted to 2018 IEEE International Conference on Robotics and Automation. Received September 15, 2017.

cues as a similarity metric on 2D images for tracking a hand and an object in interaction. This objective was minimised using particle swarm optimization (PSO) over the combined state space. This concept was later applied to data from depth sensors by Schmidt *et al*. [7] which used the signed distance function (SDF) as the similarity metric and gradient-based Gauss-Newton optimisation to minimise this objective.

Pauwels *et al*. [8] simplify articulated tracking as a 6D pose estimation problem given joint sensing and an initial camera pose. After articulating the manipulator according to the sensed joint positions, the manipulator is assumed rigid and fitted to the depth observation.

Generative model-fitting methods can be extended to track multiple objects in parallel and allow hypotheses rejection by applying kinematic and physical constraints [2]. However, these methods share similar properties and disadvantages with iterative closest point (ICP) algorithms. Their similarity metric is typically dependent on local visual features such as edges and gradients and hence can suffer from local minima and need to be initialised close to the optimal solution.

Discriminative Tracking: Meanwhile, discriminative methods learn the visual representation of a model with respect to the true state or joint configuration. This requires an extensive amount of labelled training images which show the tracked object in many different states. In this problem domain, these states are synthesised using known articulated models *a priori*.

A popular approach for depth-based tracking of articulated objects is to use simple depth probe offset features in a random forest (RF) for segmentation and keypoint localisation. This was used for human pose estimation [9] and more recently was applied to robot manipulator configuration estimation [10]. In our work we also use this type of feature and classification method, but our approach uses the raw class probability for model-fitting instead of the joint position prediction or mean-shift.

Direct regression of the full manipulator configuration has been demonstrated in [11], again using depth probe offset features. Tompson *et al*. [12] applied convolutional neural networks to depth data to detect the locations of hand keypoints on joints and to infer the joint configuration from inverse kinematics.

Hybrid: Hybrid tracking methods use a combination of generative and discriminative methods, so as to augment model-fitting with discriminative information.

The detection of fingertips was used by Tzionas *et al*. [13] and Taylor *et al*. [4] in their objective function to guide optimisation towards the optimal hand pose. In our work we propose to instead rely on a full segmentation of the image to prevent cases where these specific keypoints are occluded, e.g. when reaching behind an object.

Our work is similar to Sridhar *et al*. [5] and Krejov *et al*. [6], where a RF segmentation of depth images was used to support model-fitting of a human hand. Compared to the Gaussian volumetric approximation in [5] we use the full pixel-wise data-to-model association and a more realistic mesh model of the robot. Our approach extends [6] to cases with additional objects.

Finally, approaches which simultaneously track hands and objects typically rely on the knowledge specific to the object of interest such as colour (Sridhar *et al*. [14]) or shape (Schmidt *et al*. [2]). In our proposed tracking approach we aim to track the manipulator generally without knowledge of the object of interest, relying only on the 3D model of the manipulator. Specifically we do not require a volumetric representation of the object nor any specific properties of the object to enable manipulator tracking near occlusions.

III. Proposed Method

A. Augmenting the Signed Distance Function

Model-fitting approaches rely on the minimisation of an objective function $e(\cdot)$ which contains a term for the discrepancy between estimated and observed state, as well as other criteria which impose physical or kinematic constraints. For depth-based model-fitting, the truncated signed distance function has been commonly used as the metric when minimising data-to-model discrepancy.

The signed distance function, $\text{SDF}(\mathbf{x}) : \mathbb{R}^3 \mapsto \mathbb{R}$, of a rigid 3D model provides the shortest Euclidean distance of a given data point $\mathbf{x}$ to the surface of the model mesh, and is positive outside the model and negative inside the model. For articulated models, the SDF can be piecewise locally defined as SDF_i for all tracked parts $i \in [0, \dots, M]$. To minimise the data-to-model distance, we first need to assign each data point to one of the model parts.

Without knowledge of the true identity of a data point, prior approaches, such as [7], have assigned it to the closest SDF_{i^*} using

$$i^* = \arg\min_{i \in M} |\text{SDF}_i(\mathbf{x})| \quad . \tag{1}$$

The optimal pose $\theta^* \in \mathbb{R}^{6+N}$ of an articulated object is then the θ which minimises the data-to-model error when transforming each SDF_i according to the kinematic chain articulated by the N joints, and its 6D pose.

So as to minimise the huge state space of articulated tracking, it is common to use iterative gradient-based approaches such as the Gauss-Newton algorithm initialised close to the true solution. The gradient of the SDF with respect to θ is based on a temporary association between the data and model parts which is re-evaluated with each iteration. This data association criteria (minimal distance) is the same as the objective function which reinforces incorrect data associations and can lead to irreversible tracking failure.

We propose to instead replace the implicit data association in equation 1 by an explicit association using a discriminative pixel-wise classifier to provide a class probability distribution $p(c|\mathbf{f})$ per class c, given a feature vector $\mathbf{f}$, computed per pixel in the depth image I.

A data point is then explicitly assigned to the SDF_i of the part with the highest class probability

$$i^* = \arg\max_{i \in M} p(c = i|\mathbf{f}) \quad . \tag{2}$$

Preprint submitted to 2018 IEEE International Conference on Robotics and Automation. Received September 15, 2017.

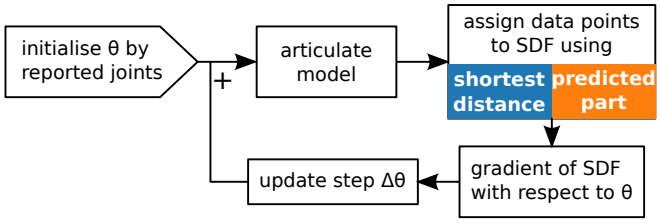


Fig. 2: Flow chart of iterative pose optimisation using either DA-SDF (shortest distance) or DA-RF (predicted part).

In what follows we refer to implicit data association using the shortest SDF distance as DA-SDF (our baseline approach), and refer to our proposed approach which uses explicit data association from pixel-wise classifications as DA-RF.

After carrying out data association, DA-SDF and DA-RF rely on the same Gauss-Newton optimisation shown in Figure 2, to minimise the data-to-model distance. After initialising the optimisation once at the reported robot state, the algorithm iteratively converges to a minimum starting from the solution to the previous iteration. Compared to [5], we evaluate only one hypothesis at a time but we use the gradients for all pixel-wise associations for the optimisation.

B. Generating Training Data for Pixel Classification

To obtain a sufficiently large set of labelled training data, we synthesise depth images using the Z-buffer of an OpenGL renderer. Each part of the robot is associated to a dedicated class and its volumetric appearance is represented by a mesh. We do not add any sensor-specific noise. Importantly, to train an occlusion robust segmentation we do not add a specific occlusion object class but instead sample pixel-wise occlusions during the feature generation phase.

To generate the training data, the robot model is articulated using a set of joint configurations which provide good coverage of the expected range of manipulation poses. We first sample poses for the palm in the task space and then validate each pose using inverse kinematics (IK). The resulting (arm) joint configurations are combined with sampled finger articulations. For each palm pose, we sample (i) a position within the camera frustum in a distance range of $[0.5, 1.5]\mathrm{m}$, (ii) axes of the rotation matrix such that the palm-face is in the direction of typical grasping. These IK solutions contain self-occlusions of the manipulator by arm segments. We obtain 4477 valid palm configurations which are each further articulated by four discrete finger grasping states (between fully opened and closed), resulting in a total of 17908 labelled training images.

C. Training

1) Pixel-Wise Segmentation of Robot Parts: To train the classification random forest (RF) for the task of pixel-wise labelling of depth images we use the depth probe offset features presented in [9]:

$$d_\Theta(I, \mathbf{x}) = d_I\left(\mathbf{x} + \frac{\mathbf{u} \cdot f}{d_I(\mathbf{x})}\right) - d_I\left(\mathbf{x} + \frac{\mathbf{v} \cdot f}{d_I(\mathbf{x})}\right) \quad . \quad (3)$$

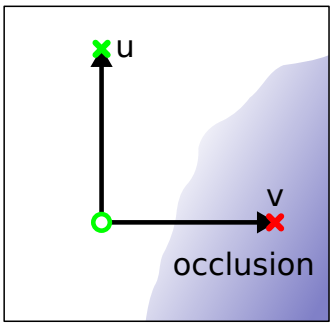


Fig. 3: Probing near the borders of occlusions (blue area) with a feature configuration $\Theta : \{\mathbf{u}, \mathbf{v}\}$. Offset $\mathbf{v}$ is probing an occluded pixel. During training, a surrogate depth value will be simulated at this location in the image which generates a different feature response.

The function $d_I(\cdot)$ gives the depth value at the queried pixel location $\mathbf{x}$ of a depth image I. The offsets $\{\mathbf{u}, \mathbf{v}\} \in \mathbb{R}^2$ are randomly sampled relative to the reference pixel $\mathbf{x}$ in world space before the feature generation and stored as one feature configuration Θ. The scaling by the focal length f allows us to use the same feature configuration independent of the distance from the camera. The feature response $d_\Theta(I, \mathbf{x})$ is then the difference of the depth at the two offset locations.

From a set of 1000 feature configurations Θ, 32 are randomly chosen each time a split node of a decision tree is optimised during training. Our RF contains 30 randomised decision trees that are trained to maximum depth. So as to achieve an equal coverage of robot parts, we sample 10 pixels per image and robot part to prevent unbalanced classes.

For simplicity, we train and test without background data which we set to the constant value of $3\mathrm{m}$. It has been demonstrated [11] that an additional pre-pended RF stage can be used for foreground/background segmentation.

2) Occlusion Sampling: If a RF is only trained on the parts of a robot, this usually results in an over-confident classification of unseen data, such as occlusions or objects, as parts of the robot. Doing this would distract the data association of the model fitting stage by assigning model parts with irrelevant data and drawing the SDF optimisation away from the true configuration. We address this problem by training the random forest with generic and randomized occlusions so as to reduce the confidence of predictions in the area of occlusions. The effect of this is that the RF becomes less confident when classifying occlusions as robot parts. These less confident classifications can then be rejected using a threshold on the class probability, with only the more confident data associations then used for model fitting.

At training time, this confidence can be shaped by randomly sampling occlusion pixels when generating the feature responses. Each time a probe offset (eq. 3) $\mathbf{u}$ or $\mathbf{v}$ accesses a pixel of the original synthetic training image, with a certain probability it is marked as accessing an occluding pixel (Figure 3).

We temporarily replace the depth value at the probe offset that has been marked as occluded by a simulated occlusion

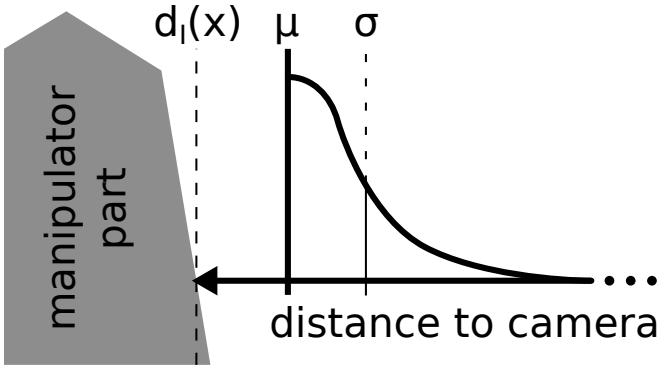


Fig. 4: Adding robustness to occlusion: During training, the original depth value $d_I(\mathbf{x})$ of a robot part (grey) is replaced by a depth value which simulates occlusion.

depth value. The reference pixel keeps its label but receives a different response from the same feature configuration. In this manner, the RF is forced to learn a certain variance of the feature response resulting from nearby occlusions.

The simulated occlusion depth value is drawn from a half-normal distribution (Figure 4). The half-normal distribution has been chosen because it has no support for points behind the farthest occluded distance and allows the occluder to have a varying shape. A depth value $d_I(\mathbf{x})$ of an occluded probe is replaced in eq. 3 by

$$d_I(\mathbf{x}) = d_I(\mathbf{x}) - |\mathcal{N}(\mu, \sigma)| \tag{4}$$

with

$$\begin{aligned} \mu &= \mathcal{U}(0, \mu_{max}) \\ \sigma &= \mathcal{U}(\sigma_{min}, \sigma_{max}) \end{aligned}$$

where μ is sampled per image and σ is sampled per probe. μ_{max}, σ_{min} and σ_{max} are held constant when training.

We will refer to our extended RF training with occlusions as DA-RF-OCCL.

IV. Evaluation

Platform: We tested the proposed approach using a KUKA LWR4 7DOF arm with a Schunk SDH2 7DOF hand (Figure 5). The hand contains 3 fingers with 2 joints each and an additional joint that allows two of the fingers to rotate around their longitudinal axis. Depth images were collected by an Asus XTION PRO Live structured light sensor. Since the depth sensor is not part of the kinematic chain, its pose in the robot frame is estimated using an AprilTag [15] mounted on the base of the robot.

During our experiments, we only track a subset of the robot's links which contains the hand and the last 4 links of the arm. The tracked state therefore consists of the 6D pose and 10 joints. The camera pose is chosen such that the arm enters the scene from the right side of the image and it is held static during a sequence.

Error Metrics: The tracked state for the baseline algorithm (DA-SDF) and variants of our approach (DA-RF with and without occlusion sampling) is compared to the robot state

Fig. 5: KUKA LWR 4 (7DOF) with Schunk SDH2 (7DOF) mounted on table with AprilTags for camera pose estimation.

as reported by joint sensing. The reference pose of a frame is obtained by forward kinematics using the reported joint positions.

We define pose tracking error T_{err} as the transformation that needs to be applied on the estimated pose T_{est} to obtain the reference pose T_{ref} in the camera frame. Decomposing $T_{err} = T_{est}^{\top} T_{ref}$ into its translation part $\mathbf{t}_{err}$ and rotation part R_{err}, the magnitude of the position error p_{err} and orientation error o_{err} are defined as

$$p_{err} = \|\mathbf{t}_{err}\|_2 \tag{5}$$

$$o_{err} = \left| cos^{-1}\left(\frac{\mathrm{Trace}(R_{err}) - 1}{2}\right)\right| \quad . \tag{6}$$

A. Experiment 1: Discriminative Tracking

In this experiment, we articulated the palm and the fingers of the manipulator without any external occlusions. This is to show the general ability of our approach to track palm pose and finger motions. Since no occlusions are present, we trained the RF without random occlusion sampling. Data points and model parts are associated with the class of highest probability.

1) Palm Pose Tracking: Figure 6 shows that tracking with RF classification (DA-RF) can achieve similar tracking performance as the simple model-fitting to closest points (DA-SDF). Given an optimal observation and initialisation, tracking by model-fitting does not benefit from discriminative

Preprint submitted to 2018 IEEE International Conference on Robotics and Automation. Received September 15, 2017.

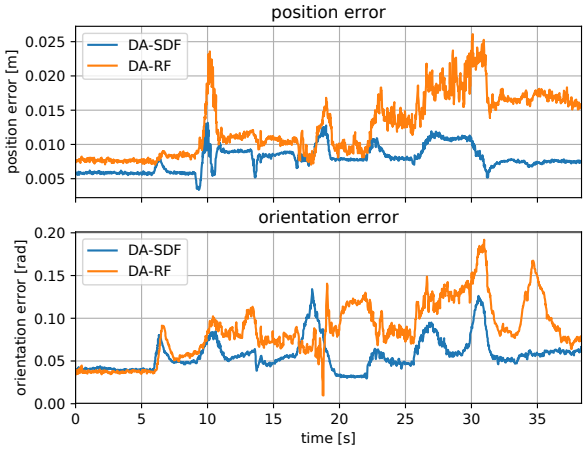
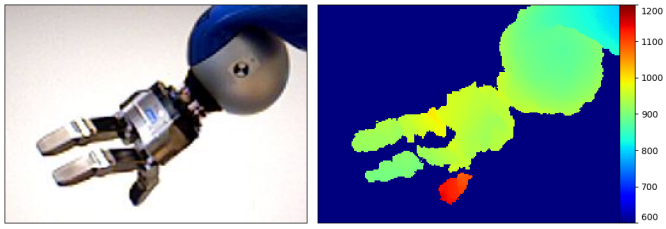


Fig. 6: Experiment 1: Palm pose tracking error. Average: DA-SDF: 0.7 ± 0.2cm, 0.06 ± 0.02rad; DA-RF: 1.3 ± 0.4cm, 0.09 ± 0.03rad.



(a) colour (b) depth

Fig. 7: Experiment 1: Static pose showing all fingers for convergence analysis.

information. These conditions are however unrealistic in scenarios like grasping, which require more robust visual tracking approaches that can deal with distractions.

2) Convergence of Optimisation: To evaluate the convergence properties of Gauss-Newton optimisation using both data association approaches, we selected a static palm pose with all fingers visible (Figure 7). We initialised the optimisation with a perturbation applied to the true palm pose. 100 of these pose perturbations were randomly sampled within the range of ± 0.1m per coordinate and $\pm \frac{\pi}{2}$rad per Euler angle.

The estimated palm pose error after converging with 500 iterations is reported in Figure 8 with cumulative histograms. Using DA-SDF as objective results in many local minima, which are located far away from the original reference pose. Only 25% of DA-SDF trials converge to palm poses with errors less than 1.5cm and 0.3rad. The DA-RF objective has less local minima and 75% of trials converge to poses within the same error bounds.

The example failure case in Figure 9 demonstrates the need to explicitly associate data points to model parts (DA-RF) to avoid local minima caused by implicit data association (DA-SDF).

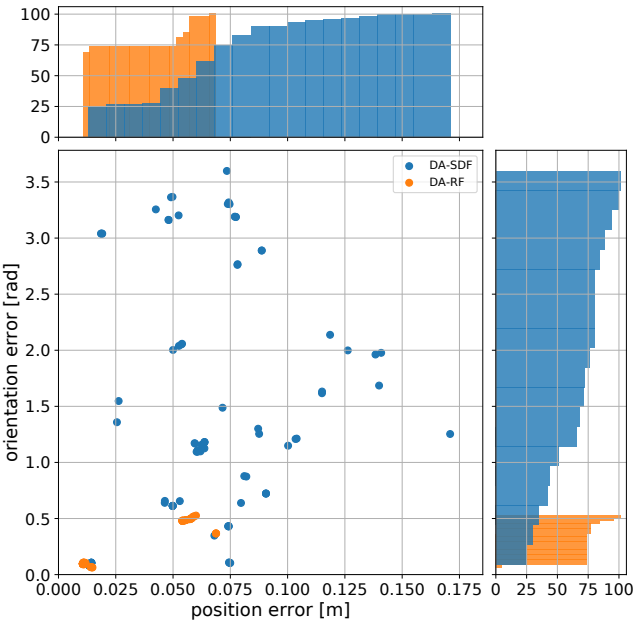
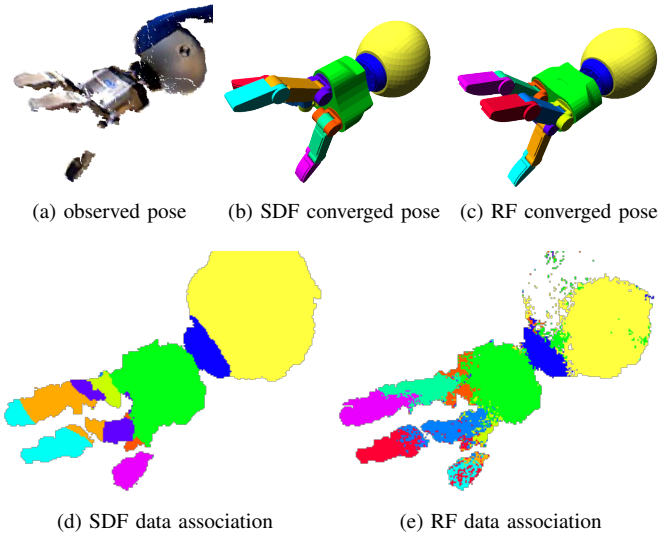


Fig. 8: Experiment 1: Palm pose error after 500 iterations converging from a perturbed initial pose. The DA-SDF objective has many local minima and causes most trials to converge to poses more than 1.5cm and 0.3rad away from the true reference pose.



(a) observed pose (b) SDF converged pose (c) RF converged pose

(d) SDF data association (e) RF data association

Fig. 9: Experiment 1: Converged poses and their data association. DA-SDF iteratively assigns the thumb (cyan) to both fingers (d), resulting in converging to local minima (b). The segmentation by the RF (e) correctly distinguishes the fingers and the thumb and makes the optimisation converge to a correct pose (c).

Preprint submitted to 2018 IEEE International Conference on Robotics and Automation. Received September 15, 2017.

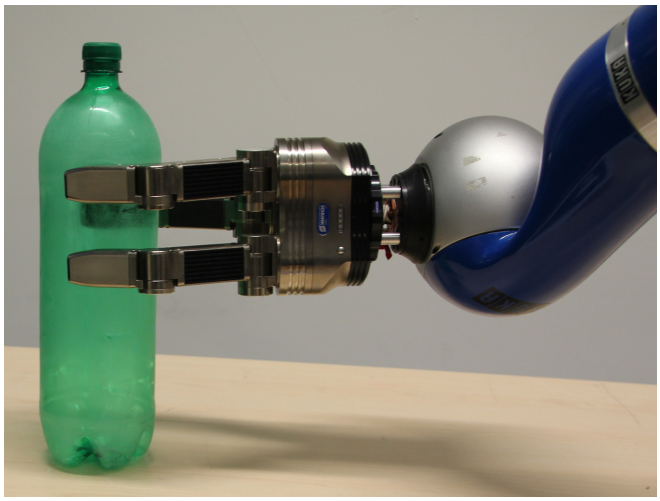


Fig. 10: Experiment 2: Grasping and manipulating a bottle.

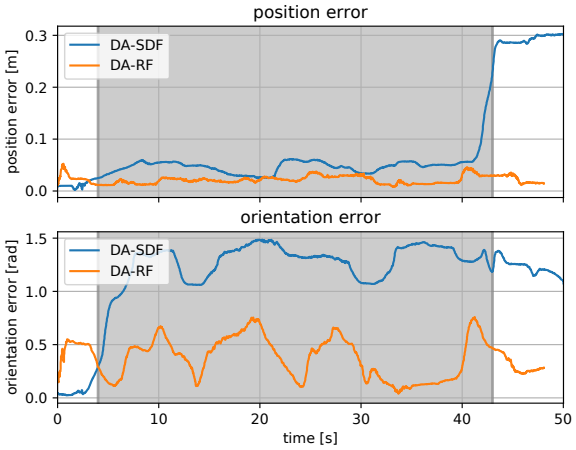
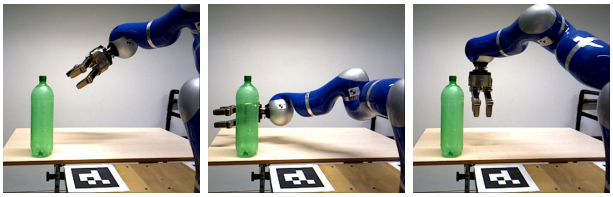


Fig. 11: Experiment 2: Palm pose estimation when grasping and moving the bottle (grey shaded phase). DA-SDF tracker is biased as the manipulated object draws the palm away from its true position. Average tracking error: DA-SDF: $8.3\pm 9\text{cm}$, $1.15 \pm 0.39\text{rad}$, DA-RF: $1.5 \pm 0.6\text{cm}$, $0.2 \pm 0.15\text{rad}$.

B. Experiment 2: Grasping

A more realistic scenario is presented by the grasping task shown in Figure 10. In this scenario, the manipulator (1) approaches and grasps an object, (2) lifts and moves the object, (3) places it back on the table and (4) moves away from the object. The baseline approach (DA-SDF) wrongly attaches the mis-tracked manipulator to the data corresponding to the object when after the initial grasp (Figure 1b). This causes the palm pose estimate to be biased during subsequent tracking (Figure 11, $t > 4\text{s}$). When retracting the hand, the tracked manipulator remains associated to the object and tracking cannot recover ($t > 40\text{s}$).

By comparison our approach (DA-RF) tracks the palm pose accurately throughout as parts of the manipulator are correctly classified during the grasping and therefore provides the correct data-to-model association.



(a) initial (b) occluded (c) unoccluded

Fig. 12: Sample images from experiment 3. Tracking is initialised at an unoccluded configuration (a), the hand moves behind the green bottle (b) and returns to an unoccluded configuration (c).

C. Experiment 3: Tracking in the Presence of Occlusions

We evaluate our main contribution in an experiment where the manipulator is occluded by an object in the near ground. This is different from the previous experiment as the occluding object is segmented into manipulator parts and the actual part is hidden and must not be associated to the occluder.

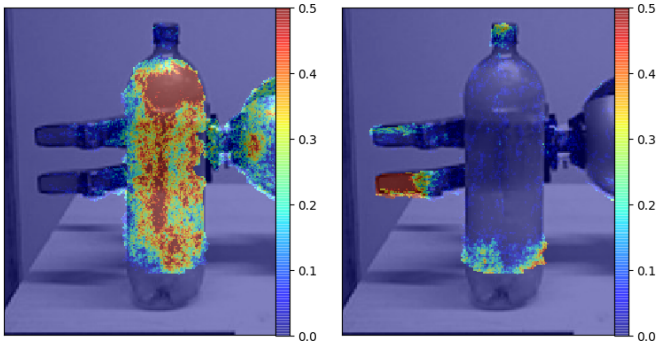
In this sequence, we initialise the robot in a state where none of its parts are occluded. The manipulator is then moved behind a green bottle such that it occludes the palm and fingers during movement so as to investigate the ability to fit the model to partial observations. The manipulator later moves back to a non-occluded configuration to demonstrate the ability to recover from tracking errors. Characteristic states of this sequence are shown in Figure 12.

1) Improved Data-Association Through Occlusion Training: We wish to be implicitly robust to unknown objects and do not want to be rely on object tracking (simplified or other). To overcome the distraction of the occluding green bottle, we train DA-RF-OCCL by adding a random sampling of occluding pixels as described in III-C.2. We sample simulated occlusions with a probability of 0.15, and replace the original depth value from a half-normal distribution with $\mu = \mathcal{U}(0, 0.1)\text{m}$ and $\sigma = \mathcal{U}(0.05, 0.15)\text{m}$.

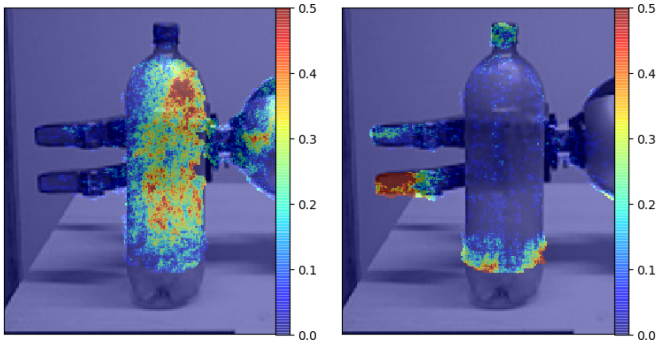
As shown in Figure 13, this random occlusion sampling decreases the probability of incorrect assignments (bottle pixels as a palm part), while the visible finger tip keeps most of its confidence. This improves model-fitting in that there are less gradients from bottle pixels that move the actual palm away from its original position.

Since we can treat the acceptance and rejection of data associations via thresholds as a binary classification problem, we can evaluate both RF classifier variants (DA-RF, DA-RF-OCCL) given the true segmentation of the object. The average Precision-Recall curve in Figure 14 shows that, given a constant threshold, our proposed training approach (DA-RF-OCCL) improves classification and hence the data-to-model association during this test sequence.

We found that the maximum depth probe distance is an important parameter that effects the model-fitting performance in presence of occlusions. For small manipulator parts like fingers, a short probe distance is important. This is visualised in Figure 15a where a large maximum probe distance (15cm)



(a) without occlusion sampling



(b) with occlusion sampling

Fig. 13: Class probabilities for palm and finger. Training without occlusions (a) results in a high classification confidence for wrongly assigning the occluded palm to the bottle and assigning the finger tip to the robot's actual finger tip. After introducing random occlusions during training (b), we can reduce the confidence of assigning bottle pixels to the robot palm but keep the high confidence of the finger tip classification.

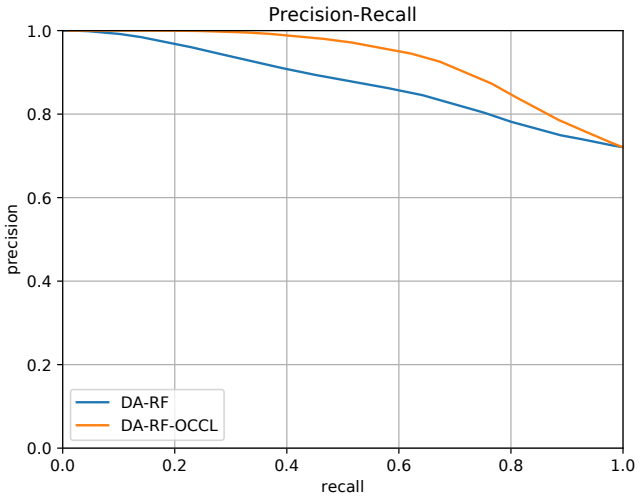
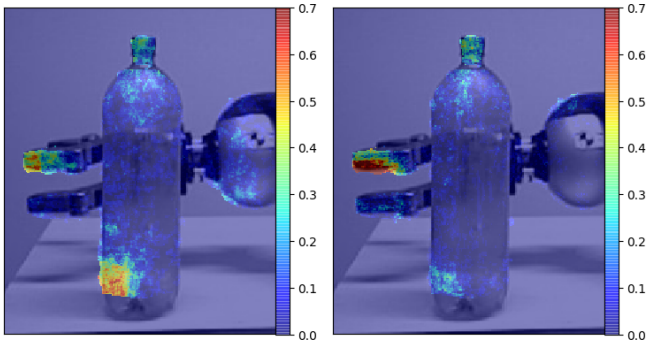


Fig. 14: Precision-Recall curve for the data association averaged over all images of Experiment 3. DA-RF: training without occlusions (AUC: 0.86), DA-RF-OCCL: training with occlusions (AUC: 0.89).



(a) probe distance: 15 cm (b) probe distance: 5 cm

Fig. 15: Class probability of finger tip for different offset distances. By reducing the probe offset distance and providing more local information, we can move probability from the bottle corner (a) to the true finger tip location (b).

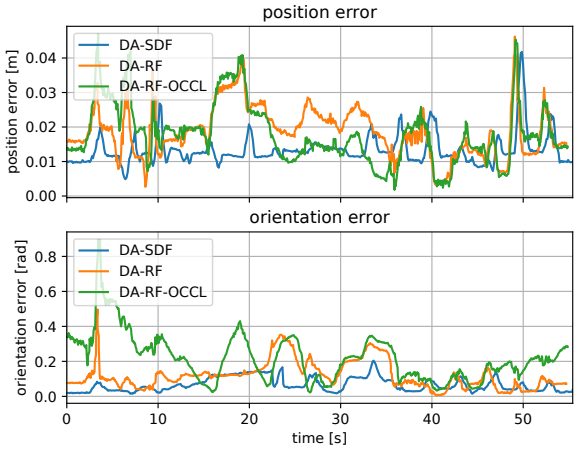


Fig. 16: Experiment 3 *without occlusions*: Palm pose tracking error after object removal. Average: DA-SDF: 1.3 ± 0.4cm, 0.07 ± 0.04rad; DA-RF: 1.9 ± 0.7cm, 0.12 ± 0.08rad; DA-RF-OCCL: 1.8 ± 0.8cm, 0.22 ± 0.14rad.

results in similar finger tip probabilities for pixels on the bottle corner and the actual finger. By enforcing learning only from local information (5cm) we can shift probability from the bottle to the actual finger tip (Figure 15b).

2) Baseline: Tracking with Known Object Pixels: As a baseline, we first use simple colour segmentation to remove the green bottle leaving only the pixels corresponding to the arm and hand (albeit with missing pixels). The idea being that this example can provide a baseline for what could be achieved when trying to be robust to a more complex unknown distractor object. The tracking error is reported for the DA-SDF, DA-RF and DA-RF-OCCL in Figure 16.

3) Occlusions: Pose Tracking Performance: Finally, Figure 17 shows the pose tracking error for DA-SDF, DA-RF and DA-RF-OCCL when rejecting pixels with a probability of less than 0.55. There is a clear difference in performance after $t > 18$s when the manipulator moves towards the bottle.

Preprint submitted to 2018 IEEE International Conference on Robotics and Automation. Received September 15, 2017.

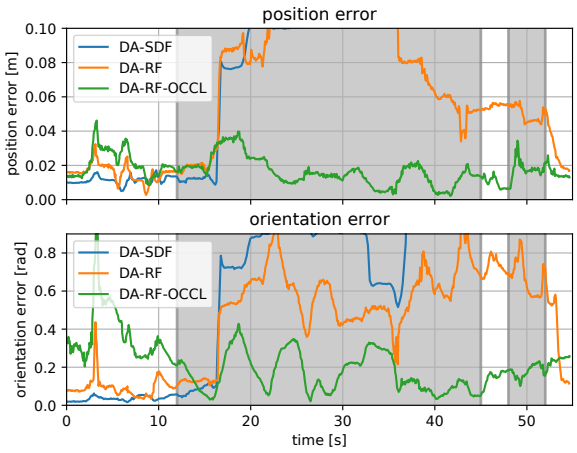


Fig. 17: Experiment 3 *with occlusion*: Palm pose tracking error during movement close to occlusion (grey shaded phase). Average: DA-SDF: 10.4 ± 8.6cm, 0.8 ± 0.63rad; DA-RF: 6.8 ± 5.1cm, 0.44 ± 0.26rad; DA-RF-OCCL: 1.7 ± 0.8cm, 0.22 ± 0.14rad.

With DA-SDF, the model is fitted to the occluding object as it cannot distinguish between parts of the robot and the bottle. By assigning irrelevant points to the manipulator, DA-SDA finally diverges and cannot recover. A similar behaviour can be observed for DA-RF, since those pixels of the bottle that have been classified as palm (Figure 13a, left) distract the tracking.

Meanwhile, by rejecting these pixels, DA-RF-OCCL is able to track with a similar performance as without occlusions (Figure 16). Without data association of finger classes, tracking relies on the visible parts of the arm until the hand becomes fully visible again.

V. Conclusion

In this work we presented an discriminative model-fitting approach for depth-based tracking of articulated objects in the presence of distracting visual information. The approach is based on the explicit pixel-wise association of data points to model parts.

In our experimental analysis we were able to avoid local minima arising from distractions in common tracking objectives. The random sampling of unspecified occlusions during training enabled us to reject less confident data-to-model associations and provides a way of tracking partially visible manipulators.

At present, our approach does not explicitly provide estimates for occluding pixels, e.g. we can only indirectly infer occlusions from low class probabilities. In future work, we propose to use multi-label classification so as to classify robot parts and occlusions so that we can independently access the occlusion probability of a pixel and also which part it is occluding.

The Gauss-Newton approach only tracks a single state hypothesis provided by the single gradients of the data-to-model association per pixel. To make use of the full class probability distribution per pixel, we propose to use a global optimisation method, such as particle swarm optimisation [3], to track multiple hypotheses in parallel. Finally, we note that most articulated tracking approaches only make use of depth information, although colour can provide a much stronger cues in particular for small parts such as fingers.

Acknowledgement

C. Rauch is supported by Microsoft Research through its PhD Scholarship Programme. M. Fallon is supported by a Royal Society University Research Fellowship.

References

[1] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Full DoF tracking of a hand interacting with an object by modeling occlusions and physical constraints," in *2011 International Conference on Computer Vision*, Nov 2011, pp. 2088–2095.

[2] T. Schmidt, K. Hertkorn, R. Newcombe, Z. Marton, M. Suppa, and D. Fox, "Depth-based tracking with physical constraints for robot manipulation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 119–126.

[3] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, D. Freedman, P. Kohli, E. Krupka, A. Fitzgibbon, and S. Izadi, "Accurate, robust, and flexible real-time hand tracking," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '15. New York, NY, USA: ACM, 2015, pp. 3633–3642.

[4] J. Taylor, L. Bordeaux, T. Cashman, B. Corish, C. Keskin, T. Sharp, E. Soto, D. Sweeney, J. Valentin, B. Luff, A. Topalian, E. Wood, S. Khamis, P. Kohli, S. Izadi, R. Banks, A. Fitzgibbon, and J. Shotton, "Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 143:1–143:12, July 2016.

[5] S. Sridhar, F. Mueller, A. Oulasvirta, and C. Theobalt, "Fast and robust hand tracking using detection-guided optimization," in *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2015.

[6] P. Krejov, A. Gilbert, and R. Bowden, "Guided optimisation through classification and regression for hand pose estimation," *Computer Vision and Image Understanding*, vol. 155, pp. 124 – 138, 2017.

[7] T. Schmidt, R. Newcombe, and D. Fox, "DART: dense articulated real-time tracking with consumer depth cameras," *Autonomous Robots*, vol. 39, no. 3, pp. 239–258, 2015.

[8] K. Pauwels, V. Ivan, E. Ros, and S. Vijayakumar, "Real-time object pose recognition and tracking with an imprecisely calibrated moving RGB-D camera," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2014, pp. 2733–2740.

[9] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR 2011*, June 2011, pp. 1297–1304.

[10] J. Bohg, J. Romero, A. Herzog, and S. Schaal, "Robot arm pose estimation through pixel-wise part classification," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 3143–3150.

[11] F. Widmaier, D. Kappler, S. Schaal, and J. Bohg, "Robot arm pose estimation by pixel-wise regression of joint angles," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 616–623.

[12] J. Tompson, M. Stein, Y. LeCun, Y.un, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 5, pp. 169:1–169:10, Sept. 2014.

[13] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing hands in action using discriminative salient points and physics simulation," *International Journal of Computer Vision*, vol. 118, no. 2, pp. 172–193, 2016.

[14] S. Sridhar, F. Mueller, M. Zollhöfer, D. Casas, A. Oulasvirta, and C. Theobalt, *Real-Time Joint Tracking of a Hand Manipulating an Object from RGB-D Input*. Cham: Springer International Publishing, 2016, pp. 294–310.

[15] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2011, pp. 3400–3407.