# Folding Paper with Anthropomorphic Robot Hands using Real-Time Physics-Based Modeling

Christof Elbrechter, Robert Haschke, Helge Ritter

*Abstract*—The ability to manipulate deformable objects, such as textiles or paper, is a major prerequisite to bringing the capabilities of articulated robot hands closer to the level of manual intelligence exhibited by humans. We concentrate on the manipulation of paper, which affords us a rich interaction domain that has not yet been solved for anthropomorphic robot hands. Robust tracking and physically plausible modeling of the paper as well as feedback based robot control are crucial components for this task. This paper makes two novel contributions to this area. The first concerns real-time modeling and visual tracking. Our technique not only models the bending of a sheet of paper, but also paper crease lines which allows us to monitor deformations. The second contribution concerns enabling an anthropomorphic robot to fold paper, and is accomplished by introducing a set of tactile- and vision-based closed loop controllers.

# I. INTRODUCTION

Complex bi-manual coordination is a routine capability for humans and is achieved through many years of practice with a large variety of different objects and situations. The availability of anthropomorphic robot hands poses the challenge of realizing similar capabilities on future robots. Besides being a fascinating challenge for basic research, the realization of dexterous manipulation with anthropomorphic robot hands can make future robots more versatile and useful when dealing with the numerous objects in household environments that have been primarily designed to be used and manipulated by human hands. It also can contribute to making future prosthetic hand devices easier to control for their wearers by providing them with a considerable part of "autonomous interaction intelligence" to carry out the many tasks that become disturbingly complex when the dexterity of our natural hands is no longer available.

Due to the high complexity of the required coordination of vision, touch and motor action in an actuator system with typically several tens of degrees of freedom, most current work is focused on the manipulation of rigid objects, often with rather regular shapes, such as spheres, cylinders or brick-shaped blocks, which allows for classical planing approaches. However, to cope with the objects found in a typical household, such as food items, clothing, laundry, or articulated tools, requires us to extend these approaches to make them able to deal with non-rigidity and objects that can change their geometry during handling.



Fig. 1. Two Shadow Dexterous Hands folding a sheet of paper. The paper is tracked from visual input for fitting a deformable soft-body-physics model, which can also reproduce crease lines

To create a systematic, yet manageable route towards the handling of such objects we have initially focused on the manipulation of deformable 2D surfaces, such as sheets of paper. Handling of such objects is relevant to several application domains and poses already an interesting subset of challenges that can pave the way to the manipulation of more general non-rigid objects: manipulation under high shape variability, the need for continuous 3D visual feedback, the involvement of elasticity, "material memory" as a result of bending and folding, and the possibility of complex construction sequences exploiting these properties.

Thus we believe that a thorough understanding of manipulation strategies for paper-like objects and their replication in multi-fingered robot hands will be a significant step towards a synthesis of the "manual intelligence" [1] that we see at work when handling non-rigid objects with our own, human hands. While we have already dealt with visually guided manipulation to enable the grasping of a flat-lying sheet of paper as a first step towards richer manipulation [2], we now wish to demonstrate a next step: enabling the real-time observation of manual paper folding sequences and the use of this capability, together with proprioception and tactile feedback, to control the necessary robot actions for paper folding under visuo-tactile feedback.

Our demonstration on a 54-DOF bimanual system (see Fig. 1), involving two anthropomorphic Shadow Hands mounted on a pair of 7-DOF Mitsubishi arms, extends our earlier work [2] in several important respects. A major novel element is an object model that combines bending and folding in a physical model driven by visual input. Another

This work was supported by the DFG CoE 277: Cognitive Interaction Technology (CITEC). The authors are with the Neuroinformatics Group at Bielefeld University, Germany. {celbrech|rhaschke|helge}@techfak.unibielefeld.de

extension is the combined use of proprioceptive and tactile feedback to complement the visual feedback with important contact information for guiding the different movement phases. Finally, we improved the robustness of the vision framework and created two new hand controllers to enable folding.

The paper is organized as follows. In Sec. II, an overview of related work is provided. Sec. III introduces the used physical model and how it models memorized deformations and creases. In Sec. IV we focus on the visual tracking of the paper model by introducing a new control law for linking the model to the current observation. The capabilities of the modeling and tracking frameworks are demonstrated by tracking a sheet of paper that is folded into a paper airplane by a human. After giving a concise system overview in Sec. V, the robotic folding experiment is presented in Sec. VI. Finally, Sec. VII, summarizes and discusses the results and provides an outlook for future work.

#### II. RELATED WORK

Robotic paper manipulation has often been addressed from the perspective of origami folding. Balkom and Mason's work on "Robotic Origami Folding" [3] provides a fundamental basis for robotic paper manipulation. However, Balkom and Mason used a robot that was especially designed for such folding. A specialized robot for spreading a piece of cloth was also used by Saul et. al. [4], who presented a mathematical model of the cloth that they used for predicting the outcome of certain actions. Folding paper with anthropomorphic robot hands gives rise to many additional and still unsolved challenges. Dexterous manipulation of deformable objects has recently been reviewed in [5], in which the discussion of the major challenges in this area was presented and a set of references which try to solve them was provided. A major statement from the review is that it is necessary to fuse all available sensory input such as force-, tactile- and visual feedback.

Mathematical simulation of paper and also origami folding has been a field of research for decades [6]. However, these models are very complex and hard to extend [7]. Mathematical models are also commonly used in computer graphics. State of the art approaches use thin shell models based on discrete differential geometry that allows for impressive physical modeling of flexible materials [8]. With the availability of free and open-source physics engines, we can bypass the low level implementation of a paper model and instead work with physical models. Additionally, these approaches are not concerned with explicitly establishing a link from real world observations to the model.

Anthropomorphic systems, able to apply certain manipulations to deformable objects have been reported only recently. Kita et al. [9] presented a robot system which was able to spread clothes. Based on dense depth information from a trinocular camera, they fit a physical model to represent the object surface. A similar system to fold towels, that focuses on the visual detection and grasping of towel corners [10] was presented by Maitin-Shepard et. al. Manipulating cloth



Fig. 2. Bending constraint structure (a) without and (b) with a defined crease line. If a crease line is defined, all bending constraints (green lines) that intersect the crease line (orange line) are deactivated by lowering their stiffness coefficient.

was also addressed by Bersch et. al [11], who presented a system for autonomous laundry folding. They use fiducial markers for estimating the deformation of a T-shirt.

Even though these approaches produced impressive results, they use a simple jaw-gripper instead of coordinating a multitude of finger joints. Furthermore, their visual detection systems cannot easily be ported to dexterous manipulation due to the presence of occlusions.

Marker aided object detection and tracking is a common technique to circumvent visual detection and registration problems. Even though there are several free marker tracking libraries available [12], [13], [14], we decided to produce our own marker detection framework because none of the available systems met our requirements in terms of speed, accuracy, false-positive rate and integrability.

There are also approaches that use common key-point detection methods such as SIFT or SURF [15] features for estimating the surface shape of deformable objects [16]. However, these approaches are only applicable to rich scenes that provide enough texture for robust feature tracking.

# III. PHYSICAL MODELING OF FOLDABLE PAPER

While humans can carry out amazing manual actions with haptic feedback alone, more complex operations, including paper folding, necessitate visually guided alignment operations and thus require substantial visual feedback to be successful. The realization of a sufficiently capable vision system is, therefore, a key element for advanced manipulation capabilities. While there exist good state-of-the-art solutions for rigid objects, the implementation of real-time vision frameworks that can track articulated or deformable objects still poses many challenges. One such challenge is the proper modeling and integration of elasticity with permanent shape changes due to strong bending and folding.

In our previous work [2] we used the Bullet physics engine [17] to model elastic bending of paper sheets during manipulation. In the following, we report on an extension of this approach to include non-elastic bending that leads to permanent folds when they becomes spatially localized along 1D lines.

As in [2], we model the paper geometry with a discrete regular 2D-grid of nodes  $n_p, p \in \{1, ..., W\} \times \{1, ..., H\}$  (in our case,  $W = 15$  and  $H = 21$ ). In the physics engine, neighbored nodes (located at grid positions p and q) are



Fig. 3. Physical paper model. (a) Creaseless model fixed at one point (red sphere). (b) Same model, but with diagonal crease line (orange line). (c) An example of memorized deformation.

connected by links cpq specifying *bending constraints*. Each constraint has two parameters: the resting distance  $d_{pq}$  of the two nodes, and a stiffness coefficient  $s_{pq}$ . Controlling the range across which nodes are connected allows us to shape the bending behavior of the sheet (for our model, we found a physically satisfactory bending behavior by providing constraints for all node pairs whose city-block distance in grid coordinates is 3 or 4, resulting in 8722 constraints (see Fig. 2a)).

We use this representation to model folds in the following way (see Fig. 3): A fold is considered to be a straight line through the model. Each bending constraint intersecting this line is modified in its parameters to create a physical behavior that mimics the mechanical characteristics of a fold. This has two components: reducing the stiffness  $s_{pq}$  to account for the weakened elasticity of the material and "memorizing" the bending action to account for plastic deformations. The latter is achieved by adapting the resting distances  $d_{\text{pq}}$  of bending constraints across the fold (see Fig. 2b).

To automatically adapt the model according to observed folds, a method to identify fold lines from visual input would be necessary. This could be done by detecting persistent deviations of the model and real sheet of paper, because such deviations typically indicate crease lines. However, in our application we are able to anticipate the location of fold lines beforehand, such that we can trigger appropriate modifications to the model programmatically, which slightly simplified the visual recognition task.

## IV. VISUAL TRACKING OF MANUAL FOLDING ACTIONS

We use the proposed paper model to visually track manual folding actions. Our vision system employs five Point Gray Flea2G fire-wire cameras with quad-VGA resolution  $(1280 \times 960)$ , along with fiducial markers printed on the paper surface to identify and locate 3D key-points. A robust feedback controller ties the motion of the model nodes to the motion of the identified key-points. Our approach, which is detailed in the following subsections, works whenever the motion of the paper is slow in comparison to the processing frame rate of 15Hz.

# *A. Marker Tracking*

We rewrote our previous marker detection framework to remove any dependencies on special marker types. The new framework (which is available with our open source computer vision library ICL [18]) allowed us to systematically compare



Fig. 4. (a) Overlay of a real image with the physical model (red lines) and the detected key-points (green crosses). Each detected marker provides five key-points, its center and its corner positions. (b) The paper model is above the current detected paper position and must be moved towards the detected key-point  $\mathbf{k}_l^w$ . The key-point's model coordinates  $\mathbf{k}_l^m$  are used to estimate the interpolated model surface position  $\mathbf{c}_l^w(\mathbf{k}_l^m)$  in world coordinates. The displacement between the nearest node positions  $n_{q_i}$  and  $\mathbf{c}_l^w(\mathbf{k}_l^m)$  is weighted w.r.t. the node distances to the key-point on the model surface in order to estimate the final node velocity updates.

a wide range of common fiducial marker designs in terms of accuracy, robustness and real-time detection performance. We found that BCH-code markers [12] were best suited for our purposes, allowing fast and robust detection, automatic error-correction and a very low false positive rate.

For the estimation of 3D key-point locations we use standard N-camera triangulation methods, i.e. only key-points that are detected in at least two cameras are used. Even though the markers are well suited for single view 3D detection, we do not rely on this because of the resulting poor z-position and x/y-orientation accuracy.

# *B. Vision-based Model Control*

An important aspect of our previous work [2] was to establish a link between the physical model and the visual input. It was shown that a simple P-controller acting on the velocities of model nodes provides very good stability. However, this approach was limited to controlling certain model nodes only. By introducing a new control algorithm (which will be outlined below), we are now able to control arbitrary model positions, which allows for decoupling the complexity of the simulation from the number of tracked key-points. This is an important step to replace the current fiducial marker detection with arbitrary, markerless key-point detection methods in the future. Fig. 4a visualizes the density and distribution of model nodes and key-points.

In order to present the control algorithm, we need to introduce some notation. We denote 3D world points by the superscript  $x^w$  and 2D model surface coordinates by the superscript  $x^m$ . At each time step, the marker tracking provides a set of key-points  ${k_l}$  that associate the currently observed world position  $\mathbf{k}_l^w \in \mathbb{R}^3$  with a point on the paper model surface  $\mathbf{k}_l^m \in \mathbb{R}^2$ . In our case,  $\mathbf{k}_l^w$  denotes 3D estimates of marker center and corner positions. The corresponding  $\mathbf{k}_l^m$ coordinates are derived from the well defined marker layout on the sheet of paper. The corresponding 3D position in the model, which is interpolated from surrounding model nodes, is denoted as  $\mathbf{c}^w(\mathbf{k}_l^m)$ . On the model side, with each node  $n_{\mathbf{q}}$  we



Fig. 5. Different steps needed to fold a paper airplane. The whole sequence is also shown in the video attachment to this paper. The images show the original images overlayed with the two faces of the model (red and blue grid), the origin of the model coordinate frame (red sphere) and manually added crease lines (orange lines). (a) The paper is folded in the middle, (b) diagonal folds are added for the wings, (c) left wing is made, (d) right wing is made, (e) final adjustment folds are made, and (f) shows the final paper airplane.

can associate fixed surface coordinates  $\mathbf{n}_{\mathbf{q}}^m = \mathbf{q} = (q_x, q_y)^\tau$ , a time-varying world position  $\mathbf{n}_{\mathbf{q}}^w$  and a targeted velocity vector  $n_q^{vel}$ .

The key idea of the control algorithm is to adapt the model such that the model position  $\mathbf{c}^w(\mathbf{k}_l^m)$  associated to a key-point  $k_l$  moves towards the observed world position  $k_l^w$ . This is accomplished by imposing a common velocity vector  $\mathbf{v}_l^w \in \mathbb{R}^3$  to the four model nodes  $n_{\mathbf{q}_i}$  surrounding the model coordinates  $\mathbf{k}_l^m$  of the key-point (see Fig. 4b). This velocity vector is calculated using a P-controller:

$$
\mathbf{v}_l^w = \lambda(\mathbf{k}_l^w - \mathbf{c}^w(\mathbf{k}_l^m)).
$$
 (1)

The gain  $\lambda$  is weighted by influence factors  $\alpha_{q_i}$  to adjust the contribution of the surrounding model nodes  $n_{q_i}$  according to their distance to the key-point in the model space as follows:

$$
\alpha_{\mathbf{q}_i} = \max(1 - \|\mathbf{k}_l^m - \mathbf{n}_{\mathbf{q}_i}^m\|, 0). \tag{2}
$$

This finally yields the following target velocities  $n_{q_i}^{vel}$  for the model nodes  $n_{\mathbf{q}_i}$ :

$$
\mathbf{n}_{\mathbf{q}_i}^{vel} = \alpha_{\mathbf{q}_i} \lambda(\mathbf{k}_l^w - \mathbf{c}^w(\mathbf{k}_l^m)) = \alpha_{\mathbf{q}_i} \mathbf{v}_l^w.
$$
 (3)

For the special case that the key-point coordinates  $\mathbf{k}_l^m$  exactly match model node coordinates  $n_q^m$ , we obtain our previous control law from [2], which moved a single node only:

$$
\mathbf{n}_{\mathbf{q}}^{vel} = \lambda (\mathbf{k}_l^w - \mathbf{n}_q^w). \tag{4}
$$

Naturally, individual velocity contributions obtained from different key-point updates are accumulated before being applied in the physics engine.

The controller gain  $\lambda$  was tuned manually ( $\lambda \approx 10$  yielded good results for a simulation step of 1/60 seconds for the physics engine, which we found to be best to achieve high simulation stability). In each single time step, the control law Eq. 3 is applied to all detected key-points in 10 iteration cycles. This enables the physics engine to move model nodes corresponding to detected key-points successively towards their target positions while smoothly interpolating invisible nodes using the constraints of the underlying physical model.

# *C. Monitoring a Manual Folding Sequence*

We demonstrate the capabilities of the detection and modeling framework with a typical folding sequence to create a "paper airplane". While the paper is manipulated, crease lines are manually added to the model and memorization of folds is triggered. Fig. 5 depicts typical key-frames of the

sequence. In Fig. 5a, a center crease line has been added to the model. The center fold is perfectly reflected by the model deformation. Also diagonal creases are approximated very well (Fig. 5b). The model is explicitly set up to memorize the current deformation, to avoid diagonal folds being lost while the paper is rotated off plane during the next steps. In Fig. 5c, the first wing is created, for which extra crease lines were added before. Here the physics engine's collision handling (self collision of the model cells as well as model to ground collision) is needed. Otherwise, the invisible parts of the model would slowly go back to their rest configuration. When the second wing is created (Fig. 5d), only a few markers remain detectable due to occlusions. While the relative configuration of the model remains overall good, it's orientation tracking is weak. This can be explained by the rotation inertia of the physical model that is not well compensated by controlling a single or just a few key-points. The most difficult part of the sequence is when the fold-angles are adjusted. Fast marker-movements and massive occlusions temporarily lead to implausible modeling results (Fig. 5e). However, once the finished paper airplane is presented without hand occlusions (Fig. 5f), the remaining detectable markers are sufficient to resume the correct tracking of the model.

## V. ROBOT SYSTEM

For the manipulation tasks, two redundant 7-DOF Mitsubishi PA-10 robot arms, each equipped with a 20-DOF Shadow Dexterous Hand, are used. For better similarity to the kinematic arrangement of human arms, both robot arms are attached in downward orientation from the ceiling. Both hands are equipped with *different* types of tactile sensors. The right hand's fingertips are equipped with a tactile sensor matrix comprising 34 tactels, providing a spatial resolution of 3mm, however at the expense of a rather limited sensitivity. On the left hand we employ PST sensors from the Shadow Robot Company. A cavity in each fingertip is used to transform contact forces into an air pressure signal, which is highly sensitive to small forces. Therefore, we make asymmetric use of the hands and employ the right hand for fixating the paper and the left hand for manipulation actions. Additionally, we added rubber finger covers to the right hand to decrease slippage of the foam-covered fingertips, and fabric covers to the left hand to decrease friction of the rubber-covered tips.

The whole robot setup is controlled by a hierarchical state machine (HSM), which coordinates numerous processes that



Fig. 6. Component communication. The camera images are processed on a single PC. The 2D detection results are fed into the 3D estimation and modeling unit that conducts multi-view 3D estimation for the marker key-points for updating the internal physical model. The current model status is published to the Active Memory component [19]. Finally, the controlling HSM module uses active memory to control the robot's hands and arms.

are distributed over several PCs using the XCF middle-ware toolkit [19] (see Fig. 6).

# VI. BI-MANUAL PAPER FOLDING BY A ROBOT SYSTEM

We now demonstrate the interaction of all components allowing a robot to bi-manually create a fold line in a standard A4 sheet of paper. This constitutes a key element for building longer action sequences that can lead to more complex paper configurations (the demonstration of such action sequences will be the subject of future work). To realise this initial step, the visual information needs to be complemented with tactile sensing from the finger tips and force information from the fingers, obtained by evaluating joint-sensor feedback from the robot hands.

Two basic hand-arm controllers serve as our building blocks for the coordination of the required actuator movements, using visual, tactile and proprioceptive feedback.

# *A. Hand-Arm Controllers*

The first controller,  $C_{\text{contact}}$ , allows force-feedbackcontrolled establishment of contact with an object. Although our robots lack explicit force-torque sensors, we achieve this by monitoring the joint sensors of the compliant Shadow Robot hand during a contact motion. As soon as the deviation of selected joint angles  $\theta$  from their relaxed reference pose  $\theta_{\text{ref}}$  exceeds a predefined threshold  $\Delta\theta_{\text{max}}$ , the contact motion of the arms is interrupted, i.e.

stop motion if 
$$
\|\boldsymbol{\theta} - \boldsymbol{\theta}_{ref}\|_P > \Delta \boldsymbol{\theta}_{max}
$$
, (5)

where  $\lVert \cdot \rVert_P$  denotes a norm weighted by a diagonal selector matrix P. Due to the passive compliance of the muscleactuated hands, joint deviations directly correspond to an applied force according to a (nonlinear) spring law. The controllable joint stiffness and the employed threshold can be used to indirectly select an appropriate contact force. Additionally, sufficiently sensitive tactile sensors can be used to detect contact. However, often they respond to restricted sensor areas only, such that contact sensitivity is restricted to certain contact situations. On the left hand, which is equipped with highly sensitive tactile sensors, we combine both jointand tactile-based contact detection.

The second controller,  $C_{\text{force}}$ , takes over as soon as contact has been established. Its task is to maintain a certain contact force on the touched surface based on the hand's tactile sensor outputs. It is realized as a simple P-controller driving the flexion of the fingers (excluding the thumb) based on the observed tactile force error:

$$
\Delta \theta_i^{\text{flex}} = \alpha_i \cdot k_P \cdot (f_{\text{target}} - f_{\text{current}}), \tag{6}
$$

where the  $\alpha_i$  denote joint-specific gain modulation factors to accomplish stronger flexion of proximal joints. Otherwise fingers would tend to curl in, thus loosing contact with the tactile-sensitive palmar surfaces of the fingertips.

Because the controller acts on hand joints only, it can be easily combined with independent arm movements tangential to the touched surface. We note an interesting integration of visual and tactile feedback: the tangential movement is controlled using visual feedback, while the motion normal to the object surface is controlled using the tactile sensors.

# *B. The Folding Sequence*

We begin by placing a sheet of paper flat on the surface of the robot's workspace. The paper needs to be moved by the robot to allow a corner to be picked up. In [2] we showed that picking up a sheet of paper lying on a flat surface is a very challenging task involving bi-manual operations. Here we follow a different strategy, shifting the paper to a suitable position for direct grasping. Humans often do this when objects are too flat to otherwise pick up. The paper is shifted by sequencing the controllers  $C_{\text{contact}}$  and  $C_{\text{force}}$  (see Fig. 7a). Once correctly positioned, the paper is fixed by the right hand and pinch-grasped by the left hand (see 7b). To realize the bending motion, the robot follows a predefined trajectory formed from several points defined relative to the current paper coordinate frame. During this motion, the paper model is modified by adding the desired crease line in order to allow a higher bending curvature of the model (see 7c and d). Otherwise, it would not be able to adapt to the strongly bent sheet of paper. Instead the lower layer, whose markers are occluded and thus cannot constrain the motion of the model anymore, would begin to unfold again. After the bending motion, the back of the left hand's finger tips are used to fixate the paper on the table.

Even though we do not use closed-loop feedback during the bending motion, we track the deformation of the paper to know where the different parts of the paper are in the next step. Here, the right hand is released in order to be able to fixate the now folded top layer of the paper onto the bottom layer. Again,  $C_{\text{contact}}$  is used here (see Fig. 7e). We are now able to generate a crease, which is performed in two sub-steps. First, a coarse swiping motion with the left hand is performed to better define the actual crease line (see Fig. 7f). This motion is technically identical to the initial shifting motion of the right hand, so a sequence of  $C_{\text{contact}}$ followed by  $C_{\text{force}}$  is used once more. However, this time, the motion is applied with the left hand whose fingertip friction is low, while the paper is fixated strongly with the right hand. For the second step, the actual crease line is estimated by the vision-system to compute the fingertip trajectory for the final



Fig. 7. Robotic paper folding sequence, also available in the video attachment to this paper. (a) The paper is shifted for pinch-grasping its corner. (b) Pinch grasp with left hand. (c) Paper is bent with left hand. (d) The center crease is inserted to allow higher curvature. The left hand temporarily fixates the paper. (e) Right hand is released and is now used to fixate both paper layers. (f) The left hand performs a swiping motion to produce a crease in the paper. (g) The left hand then performs a precise creasing motion in order to harden the crease. (h) The folded sheet of paper and its associated model are shown.

crease making motion. For higher precision, the crease line is hardened with two fingers (see Fig. 7g). Again, we derive appropriate contact points with the paper from the model. Once the crease line is created, the paper model memorizes it's deformation and can be released by the hands (Fig. 7h).

# VII. DISCUSSION AND FUTURE WORK

We proposed a method for real-time detection and physical modeling of paper. The system runs very robustly even in presence of heavy occlusions and fast paper movements. We also demonstrated that crease lines and memorized creases can be represented plausibly by altering the bending constraints of the physical model. The detection and modeling capabilities of the system were demonstrated during an interaction sequence in which a piece of paper was folded into an airplane. Utilizing our detection and modeling framework, we were able to endow a bi-manual anthropomorphic robot with the ability to fold a piece of paper in two. For this complex manipulation task, closed-loop tactile-, force- and visual-feedback controllers were devised and successfully demonstrated.

The most obvious drawback of the presented system is the necessity of having fiducial markers for key-point estimation. By introducing a new flexible control law for model updates, we have already prepared our system for a future replacement of marker detection by common general image features, such as SURF [15]. However, most sheets of paper we deal with are either blank or contain machine written text calling for methods, that do not rely on paper textures. 3D cameras such as the Microsoft Kinect are good candidates to acquire reliable 3D information at a low cost. However, since the Kinect device only provides 3D positions without corresponding 2D paper coordinates, the model update mechanism would need to be adapted. Even though modeling creases by altering bending constraints was shown to work very accurately even in situations with intersecting folds as occurred during the creation of the paper airplane, we would like to be able to define crease lines more precisely. This can be achieved by inserting new model nodes and bending constraints where a crease line intersects with the model grid. This adaption would release the necessity for a fine-grained initial model grid, because crease lines would no longer need to be approximated by a set of model cells. Currently, folds have to be added explicitly by the user, which is acceptable in case of active folding. However, it makes it very difficult to detect a divergence between intended and performed creases. A possible relaxation of this is to use a set of model hypotheses/particles, e.g. with parameters randomly

drawn around the intended one, which would enable us to select the model that best reflects the current observation. It would also be possible to derive the parameters of a crease line from the forces applied to the paper.

Using online visual feedback to estimate parameters of the motion sequence, the folding task is successful in 4 out of 5 trails. By enabling our robot to fold paper, we have provided a promising basis for more complex manipulation patterns. However, we hope to get more insights into the inherent structure of the problem of manipulating paper by tackling more difficult interaction sequences. This would enable use to create an interaction toolkit with building blocks for manipulating paper and also other deformable objects.

#### **REFERENCES**

- [1] J. Maycock, D. Dornbusch, C. Elbrechter, R. Haschke, T. Schack, and H. Ritter, "Approaching manual intelligence," *KI - Kunstliche ¨ Intelligenz*, vol. 24, pp. 287–294, 2010.
- [2] C. Elbrechter, R. Haschke, and H. Ritter, "Bi-manual robotic paper manipulation based on real-time marker tracking and physical modelling," in *Proc. IROS*, 2011.
- [3] D. Balkcom and M. Mason, "Robotic origami folding," *The International Journal of Robotics Research*, vol. 27, no. 5, p. 613, 2008.
- [4] S. Cuen Rochin, J. Andrade Cetto, and C. Torras Genis, "Action selection for robotic manipulation of deformable objects," in *Proc. Frontier Science Conference Series for Young Researchers: Experimental Cognitive Robotics, 2008*, 2008, pp. 1–6.
- [5] F. F. Khalil and P. Payeur, "Dexterous Robotic Manipulation of Deformable Objects with Multi-Sensory Feedback - a Review," *Robot Manipulators, Trends and Development, In-Teh (Eds)*, March 2010.
- [6] T. Hull, *Origami3: Third International Meeting of Origami Science, Mathematics, and Education*. AK Peters Ltd, 2002.
- [7] T. Tachi, "Simulation of rigid origami," *Origami 4*, p. 175, 2009.
- [8] E. Grinspun and A. Secord, "Introduction to discrete differential geometry: the geometry of plane curves," in *SIGGRAPH ASIA*, 2008.
- Y. Kita, E. Neo, T. Ueshiba, and N. Kita, "Clothes handling using visual recognition in cooperation with actions," in *Proc. IROS*, 2010.
- [10] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding," in *Proc. ICRA*, 2010.
- [11] C. Bersch, B. Pitzer, and S. Kammel, "Bimanual robotic cloth manipulation for laundry folding," in *Proc. IROS*, 2011.
- [12] M. Fiala, "ARTag, a fiducial marker system using digital techniques," *Proc. Computer Vision and Pattern Recognition*, vol. 2, 2005.
- [13] M. Kaltenbrunner, "reacTIVision and TUIO: a tangible tabletop toolkit," in *Int. Conf. Interactive Tabletops and Surfaces*, 2009.
- [14] D. Wagner and D. Schmalstieg, "Artoolkitplus for pose tracking on mobile devices," in *12th Computer Vision Winter Workshop*, 2007.
- [15] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Underst.*, vol. 110, Jun. 2008.
- [16] J. Pilet, V. Lepetit, and P. Fua, "Augmenting deformable objects in real-time," in *4th Int. Symp. on Mixed and Augmented Reality*, 2005.
- [17] "Bullet Library." [Online]. Available: http://www.bulletphysics.org C. Elbrechter, M. Götting, and R. Haschke, "Image Component
- Library (ICL)." [Online]. Available: http://iclcv.org [19] Bauckhage, Wachsmuth, Hanheide, Wrede, Sagerer, Heidemann, and
- Ritter, "The visual active memory perspective on integrated recognition systems," *Image and Vision Computing*, vol. 26, 2008.