



HAL
open science

Augmented Reality during Cutting and Tearing of Deformable Objects

Christoph J. Paulus, Nazim Haouchine, David Cazier, Stephane Cotin

► **To cite this version:**

Christoph J. Paulus, Nazim Haouchine, David Cazier, Stephane Cotin. Augmented Reality during Cutting and Tearing of Deformable Objects. The 14th IEEE International Symposium on Mixed and Augmented Reality, Sep 2015, Fukuoka, Japan. pp.6. hal-01184495v2

HAL Id: hal-01184495

<https://inria.hal.science/hal-01184495v2>

Submitted on 1 Sep 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Augmented Reality during Cutting and Tearing of Deformable Objects

Christoph J. Paulus*, Nazim Haouchine†, David Cazier‡, Stephane Cotin§

Inria, France
Université de Strasbourg, ICube Lab, CNRS, France

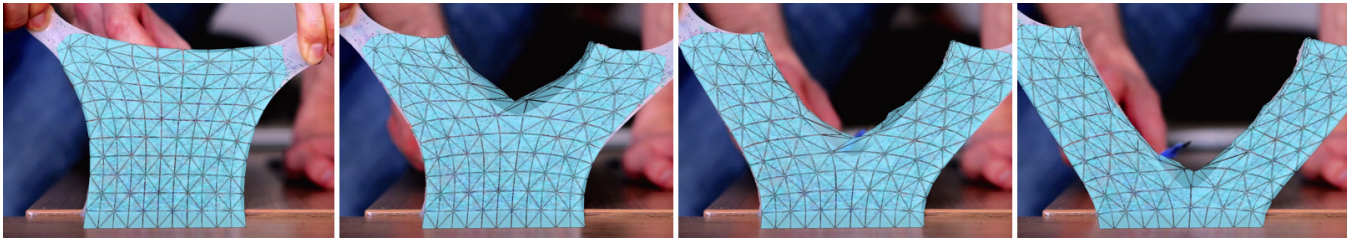


Figure 1: An augmented elastic object undergoing large deformations and topological changes. The computation of the physics-based deformation, the cut detection and the topological modification of the underlying volumetric model are all performed in real-time.

ABSTRACT

Current methods dealing with non-rigid augmented reality only provide an augmented view when the topology of the tracked object is not modified, which is an important limitation. In this paper we solve this shortcoming by introducing a method for physics-based non-rigid augmented reality. Singularities caused by topological changes are detected by analyzing the displacement field of the underlying deformable model. These topological changes are then applied to the physics-based model to approximate the real cut. All these steps, from deformation to cutting simulation, are performed in real-time. This significantly improves the coherence between the actual view and the model, and provides added value.

Keywords: Augmented Reality, Deformation, Cutting, Tearing

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling

1 INTRODUCTION

Augmented reality has shown significant promise in overcoming certain visualization and interaction challenges in various domains such as medicine, construction, advertising, manufacturing, and gaming. Despite the promise of augmented reality and its successful application to many domains, significant research challenges remain. Among these challenges is the augmentation of non-rigid structures that can undergo topological changes, such as fracture, tearing or cutting. This is for instance the case in minimally invasive surgery, which has gained popularity and became a well-

established procedure thanks to its benefits for the patient, in particular with shortened recovery times. However, this surgery remains complex from a surgical point of view, mainly because of the reduced field of view which considerably impacts depth perception and surgical navigation. The introduction of augmented reality during surgery has the potential to significantly improve these limitations [16]. Despite such recent improvements, no study has yet investigated the impact of cutting or resection actions performed during the operation. Given that these are essential steps of any surgical procedure, it is obvious that if the meshes of the underlying mechanical model are not correctly modified, significant errors are generated in the registration and consequently in the estimation of internal structures or tumor localization. The motivation of this paper is to show results address these issues.

There are very few works currently combining cutting simulation and augmented reality on deformable objects in real-time. And yet, such augmentation cannot be performed unless the actions applied onto the real object can be identified as a source of topological modification. Two directions can be investigated for this: using computer vision to detect if a “cutting instrument” is in contact with the object, or by analyzing the deformation of the underlying mesh. In this paper we will rely solely on the second approach.

2 RELATED WORKS

In this section we present previous works related to augmented reality and topological changes. Since we propose to address the issue of deformable objects, we first present an overview of non-rigid augmented reality methods. Second, previous works on mesh cutting techniques using physically-based models are presented. In the last part, we highlight our contribution.

2.1 Augmented Reality with Deformable Objects

Depending on the needs, the establishment of an augmented reality method capable of handling non-rigid objects can be seen as a non-rigid registration between the image data and the deformable model, with the aim of performing a coherent spatio-temporal overlay; or as a 3D shape recovery, with the aim to reconstruct the 3D targeted object from a monocular view. In both contexts, we can

*e-mail: christoph.paulus@inria.fr

†e-mail: nazim.haouchine@inria.fr

‡e-mail: david.cazier@unistra.fr

§e-mail: stephane.cotin@inria.fr

identify two different types of surfaces: inextensible objects such as t-shirt or papers, and extensible objects such as human organs.

Prior works have first considered inextensible surfaces where the exploitation of the underlying distance constraints is possible. Most of the time parametric geometrical models such as Radial Basis Mapping (RBF) and Free-form Deformation (FFD) are used as warp function to allow the 2D deformation to be computed [2, 20, 31, 19]. Other methods rely on learning-based approaches where the solution is computed from a representative sample of possible shapes using a dimensionality reduction process [24, 22]. Some approaches propose to solve the problem as a convex minimization of the reprojection error formulated as a Second Order Cone Programming [23] or as Quadratic Programming [25].

Although previous methods are very effective, they are not suitable for elastic objects, where the invariance of geometrical properties cannot be considered as a plausible constraint. To overcome this limitation, a closed-form solution constrained by shading information was introduced in [14] to capture stretching surface. This method assumes a Lambertian surface with a single point light source and yields good results. However, the strong assumptions on the lighting makes the method hard to generalize in all environments. Another way to describe elastic shapes is to use physics-based models relying on elasticity theory, such as [1], where a combination of Finite Element Modeling with and Extended Kalman Filter shows the efficiency of physics-based methods. In [30], a linear finite element method is used to predict the deformation. The approach described in [13] relies on the minimization of a stretching energy subject to external image constraints. The problem is formalized as a non-linear minimization that unifies geometric constraints assuming a projective camera and mechanical constraints assuming local linear deformations. In a similar way, [7] proposed to use a non-linear elastic model to augment highly elastic objects. The model is constrained by feature points acquired from a single view camera, in addition to a set of boundary points that permits to yield a sufficient constrained system. This method was applied to surgical augmented reality.

Most of the previous methods aim at recovering and augmenting 2D surfaces. Augmenting 3D objects requires a more advanced acquisition process. In [8] a stereoscopic-based visual tracking is combined with a real-time physics-based model to augment a volumetric object. The model also provides a regularization mechanism, making it more robust to tracking errors. Similar to this method, [10] proposed to use depth camera to capture the 3D deformed shape. Once the deformation is computed, the augmentation of the deformable object is performed using a constrained mass-spring system. Recently, there has been a great deal of ongoing research efforts towards the use of biomechanical models for surgical augmented reality considering the elastic organ behavior. Patient-specific biomechanical models demonstrated their relevance for volume deformation, as they allow to account for anisotropic and elastic properties of the shape and to infer in-depth structure motion [27], [9], [21] leading to more accurate visualization of vessels and tumors and therefore better decision making.

2.2 Virtual Cutting Simulation

In computer graphics, there is ongoing work on modeling methods related to cutting, fracture or tearing simulation. A comprehensive overview of techniques used for soft tissues cutting simulation is provided by [28].

For instance, the composite finite element method (CFEM) [29] addresses the issue of real-time computation by embedding a fine grid – for the visualization and collisions – into a coarse uniform hexahedral grid – for the simulation. The fine level grid is connected by links and cuts are performed by deleting these links. As soon as enough small elements are separated in a coarse element, the coarse element is duplicated and a complete separation of that

element takes place to simulate the cut. This approach allows for visually pleasing results and real-time computation. However, the elements of the coarse mesh cannot undergo partial cuts.

Another proposition – the snapping method [17] – moves the vertices of the original grid to the separation surface and disconnects the object at these vertices. This approach deals efficiently with the cut of an object, but is limited to the topology of the original grid. In order to provide an efficient simulation of cuts avoiding this weakness, several approaches propose to combine the snapping method with a local topological update – the remeshing – of the original grid. The method proposed in [18] combines the snapping method with a novel remeshing method, that is based on simple operations like a split 1-4, a flip 2-3, a general edge-flip and the split of an edge. As the remeshing method introduces less nodes than existing approaches, it is specifically interesting for the application in real-time applications.

2.3 Contributions

The main contributions of this paper are a method to detect a cut in three-dimensional soft structures by analyzing the motion of tracked surface points and the application of the detected topological changes to the model in real-time. This leads to an improved coherence between the event captured in a video and the overlaid mesh, therefore positively impacting the accuracy of the augmentation.

3 SPATIO-TEMPORAL REGISTRATION

Our method involves capturing the deformations, cuts or tears from images acquired from a monocular camera. We rely on the spatio-temporal registration proposed in [7] where acquired feature points constrain a non-linear elastic model. We first describe here the physical-based model used to translate the elastic behavior of the object. This physical model is simulated using the Finite Element Method (FEM) that uses a discretization of the shape in tetrahedral elements e . These elements are encoded in a 3D mesh with vertices in the set $V = \{v_m \in \mathbb{R}^3\}$. Then, we outline the non-rigid registration between this physical model and the image-based data that is formulated as an energy minimization problem. Thereafter, we introduce an additional constraint that effectively ensures a robust registration to image outliers.

Physically-based Model: The choice of a relevant constitutive model is essential as it determines the set of deformations we are able to capture and estimate while discriminating non-plausible material configurations induced by tracking errors. Several reports and surveys have been published to provide an exhaustive overview of state-of-the-art methods [6, 15, 26]. In that context, the Saint Venant-Kirchhoff model [26] appears to be an ideal compromise, because of its ability to handle non-linear deformations, its rotational invariance and its simplicity compared to other non-linear models so that it supports real time computations.

The strain energy W_e of a Saint Venant-Kirchhoff material estimated for each element e in the FE model is given by

$$W_e = \frac{\lambda}{2} [tr(\epsilon_e)]^2 + \mu tr(\epsilon_e^2)$$

where λ and μ are *Lamé* coefficients and can be computed thanks to the elastic parameters of the material E and ν . E is the Young's modulus and is a measure of the stiffness of the material while ν is the Poisson's ratio and estimates the compressibility of the material. Therefore the total strain energy of the medium is a function of the edge lengths of the element mesh and elastic parameters.

Identification of Features: In order to estimate the motion of the deformable object, we rely on a simple combination of the iterative Lucas-Kanade optical flow [4, 12] and the speeded-up robust features (SURF) detector [3]. This combination shows effective results tracking highly elastic objects [7, 9].

The visual tracking step yields a set of features $F = \{f_i \in \mathbb{R}^2\}$ tracked over frames in the video stream. Each feature point f_i is associated with a virtual feature point $f_i^v \in \mathbb{R}^3$ lying on the surface of the FEM mesh V with the nodes v_m (figure 2). The points f_i^v are initialized with the first frame of the video as the intersection of the line of sight – from the camera’s position to f_i – with the boundary of V . Each virtual feature is registered in an element of the FEM mesh and expressed as barycentric coordinates of its vertices.

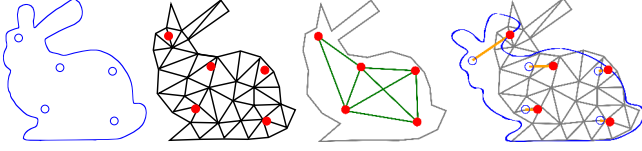


Figure 2: From left to right: A real object with the extracted features f_i ; The associated FEM mesh with the virtual features f_i^v ; The set N of pairs of neighboring virtual features; The tracked differences d_i .

Spring Forces Accumulate to Stretching Energy: In order to compare the positions of the features f_i and the corresponding virtual features f_i^v , the points f_i^v are projected onto the plane against which the video leans back, in the 3D scene. In the following, we use $f_i^* = P(f_i^v)$ to denote those projections, with the projection matrix P of the camera. As the features move in the video, they introduce spring forces between each feature f_i and its projected virtual feature f_i^* that accumulate to the stretching energy $W_S(F, V)$:

$$W_S(F, V) = \sum_i \frac{1}{2} k_i \|f_i - f_i^*\|^2$$

The parameters k_i are experimentally chosen and are in the same order of magnitude than the Young’s modulus of the deformable object.

Energy Minimization Problem: The internal elastic energy of the deformable model is given by accumulating the strain energy W_e of the elements related to a Saint Venant-Kirchhoff material:

$$W_I(V) = \sum_e W_e$$

In addition, the mechanical object is constrained by fixing nodes at predefined positions: $v_m = v_m^D, m \in B$. Finally, the deformation of the deformable model is expressed as a minimization problem between internal elastic energy and stretching energy

$$W_I(V) + W_S(F, V)$$

with $v_m = v_m^D, m \in B$. The solution of the problem is the updated set of vertices V . The positions of the virtual features f_i^v are updated using the stored barycentric coordinates of the updated V .

The stretching energy links the virtual features f_i^v to the real ones f_i . When the deformable model correctly follows the motion of the real one, the vector $d_i = f_i - f_i^*$ changes continuously in the neighborhood of f_i . This vector encodes simultaneously the euclidean distance between f_i and f_i^* and direction of the relative motion of the object and its virtual representation. In the next section, the vector d_i is used to detect potential cuts in the real object.

Ensuring Visual Tracking Robustness: To ensure a trustworthy and robust detection of cuts, unrealistically tracked features – that jump or slide in improbable directions – need to be filtered out. The underlying mechanical model used to deform the virtual object has the good property to regularize and smooth the movements of the projected points f_i^* . However, before relying on this characteristic, we filter unregular features as described in the following.

We compare the evolution of the features to determine outliers, looking for spacio-temporal discontinuities. Precisely, we analyze the evolution of the euclidean norm between features during a time step, i.e. we consider the measure $\|f_i(t + dt) - f_i(t)\|$, and determine outliers comparing to the measures’ average. If a feature f_i has been identified as an outlier, then it is neither used for the tracking nor for the detection of cutting and tearing.

Thereafter, the relative motions of the filtered features f_i and the projections of their associated virtual features f_i^* are utilized to detect topological changes in the acquired videos.

4 CUTTING AND TEARING

In this section, we describe how the emergence of a cut is detected and how this cut is replicated onto the virtual model. Our main contribution addresses the identification of topological changes by comparing the motion of the tracked object and its virtual representation. This method does not rely on computer vision techniques to detect if an instrument is generating the cut, making this approach also suitable for detecting the tearing of an object. In addition we perform the topological changes based on the finite element method described in the previous section in real-time.

Measure of Neighboring Features: Cuts or tears have two main impacts on the motion of a feature f_i and its virtual representations f_i^* . First, the distance between two real features f_i and f_j lying on different sides of the cut increases rapidly compared to the distance between other features. On the other hand, the uncut geometry of the FEM mesh does not allow the virtual object – and with that the virtual features f_i^v – to follow correctly the motion of the features f_i . This results in a divergent motion of f_i^* and f_i in the concerned area. Topological changes – cuts, tears or fracture – can thus be detected by searching outliers in the measurements: the difference d_i between f_i and f_i^* , and the relative evolution of these differences for two neighboring features. For that purpose, we introduce the set N of neighboring features – i.e. the couples $\{(f_i^*, f_j^*)\}$ such that $\delta_{ij} < r$ with $\delta_{ij} = \|f_i^* - f_j^*\|$. The radius r is closely related to the density of the features and thus to the parameters of the features extraction algorithm. The neighborhood information N is initialized with the first frame of the video. The initial distances δ_{ij}^0 – in the rest shape of the virtual object – are stored for future comparisons. Furthermore, we define the measure

$$\mu_{ij} = \delta_{ij} / \delta_{ij}^0 \cdot \|d_i - d_j\|$$

and use $\bar{\mu}$ to denote the mean value of the μ_{ij} over the set N (figure 3 (a)). The ratio $\delta_{ij} / \delta_{ij}^0$ gives us information on the relative sizes of the current distance of neighboring features f_i^* with their initial distance. An increase of this ratio indicates either an elongation of the object or a separation due to a cut or a tear. The norm $\|d_i - d_j\|$ evaluates the similarity in the motion of neighboring features.

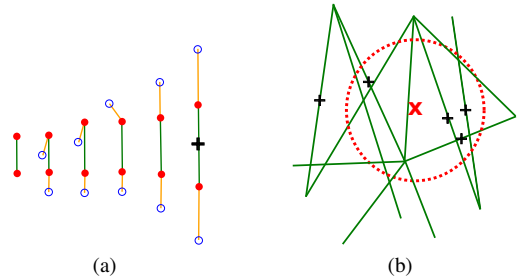


Figure 3: (a) Neighboring features: their initial position; increasing measure; an identified outlier. (b) Initialization of a cut point

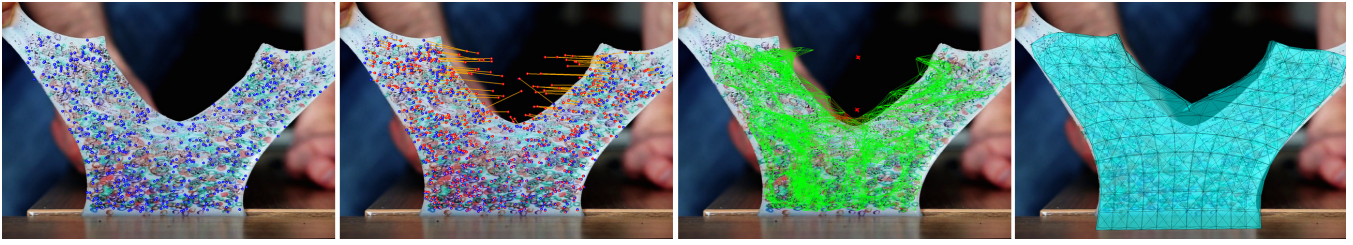


Figure 4: From left to right: raw image with image features (in blue); image features (in blue) and their corresponding point on the deformable model (in red) and the orange vector showing the distance between real and virtual features; connectivity graph between virtual features f^* (in green) with the endpoints of the cutting path (in red); resulting updated mesh overlaid on the image.

Outliers of the Measure: In a normal configuration, the deformations of the real object smoothly diffuse in the vicinity of each feature. Thus neighboring features should have motions that are similar in size and direction. In that case, the measure μ_{ij} remains small. In the case of a normal elongation, neighboring features are stretched. If the deformation stays below the mechanical strength of the virtual object, their associated virtual features controlled by the FEM mesh move the same way – with a possible latency. Thus the norm of the difference of d_i and d_j remain small and even if the two vectors have opposite directions, the measure μ_{ij} takes values that are far beyond $\bar{\mu}$. In the case of a cut or a tear, the motion of the virtual features cannot follow the real features. Therefore, the vectors d_i and d_j elongate in opposite direction and the measure μ_{ij} takes values that are far beyond $\bar{\mu}$. Outliers – i.e. pairs of divergent features – are identified as pairs such that $\mu_{ij} > \bar{\mu}\epsilon$, with a threshold ϵ that depends on the scenario.

Insertion of Cut Points: The set of outliers defines the region where a cut occurs. From this set, that evolves in time, we aim at building the separation polygons that are defined as sequences of cut points $\{c_{i_1}, \dots, c_{i_2}\}$. The polygon is initialized with $i_1 = i_2$ when the first cut point is inserted. Assuming a continuous widening of the cut, we determine that cut points can then only be inserted in the separation polygons before the first extremity c_{i_1} or after the last one c_{i_2} . The indexes of the extremities evolve by following this rule and are not detailed further.

To estimate the geometrical position of the separation, we assume that if two features diverge – i.e. are detected as outliers – the cut should be near their barycenter. As soon as the number of outliers exceeds a given threshold, a cut point c is inserted as the average of the barycenter of the outliers, as shown in figure 3 (b). Then, the pairs of features whose connecting line cuts a sphere around c are removed from the neighbors set N . Thanks to this update, the next cut points can not be in the sphere of a previously inserted cut point. The insertion threshold and the radius of the spheres are parameters that allow to balance between the precision of the cut and the robustness of the cut detection.

Extension to Cut Polygon: Finally, new cut points are inserted before or after the nearest extremity (figure 5) what makes the polygon extend to fit the widening of the cut. As the cut points are averaging the detected outliers, the built polygon may separate a feature f_i and its associated virtual feature f_i^* . If this is the case, the elastic energy $W_S(F, V)$ defined in section 3 will try to close a performed cut. We avoid this problem by the definition of a margin around the cut line figure 5 (a). The features lying inside this strip are not taken into account for the calculation of $W_S(F, V)$.

Treatment of the Boundary: The extraction of the features does not provide features that can go beyond or even on the boundary. Since the cut points – and with that the cut lines and cut planes – are positioned on the barycenter of virtual feature points, a cut polygon actually stops before the boundary of the object, preventing the occurrence of complete cuts. To resolve this shortcoming,

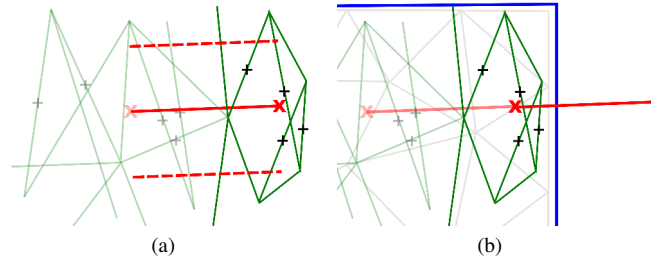


Figure 5: (a) The update of a cut point leads to a cut line (red line), the margin is displayed with dashed red lines (b) Treatment of the boundary: elongation of the current cut line over the boundary of the real (blue) and the virtual representation (grey mesh)

we check after each cut point insertion whether there are still neighboring virtual features in front of the cut. If this is not the case, the cut line is lengthened to cross the boundary of the FEM mesh and the limit of real object, as shown in image 5 (b).

Expansion to Separation Surface and Performing Topological Change: The separation polygon is finally extruded along a depth vector to define a separation surface S . This vector is either the direction of the camera or a predefined direction. The computed separation surface S indicates where the virtual object should be cut to reflect the detected cuts or tears. The detected surface S can be forwarded to any algorithm that is capable of applying the topological changes into the underlying FEM mesh. In this work, we used the combination of a local remeshing with a snapping approach described in [18].

5 EXPERIMENTAL RESULTS

In this section, we give further details on, and we demonstrate the potential of, our approach to detect topological changes by the analysis of the motion obtained from a monocular video and to incorporate these changes in a virtual volumetric deformable model used for augmented reality. Proposed ideas have been implemented using the Open Source framework SOFA [5] for the simulation of the deformation of the object. Our approach is assessed on real data using two examples of silicone cuboids.

V-Example: In the first example – displayed in figure 1 – we stretch a soft, highly extendable silicone cuboid in its length and width, while the cuboid is fixed by boundary conditions at its stem. Almost at the maximal level of elongation before tearing, we perform a vertical cut, that is preceded by a separation due to tearing. The rectangular shape changes until it resembles the letter "v" – this is why we call this example in the following the v-example. The real cuboid is 10cm in width and height and has a thickness of 8mm, while the virtual duplicate consists of 288 degrees of freedom

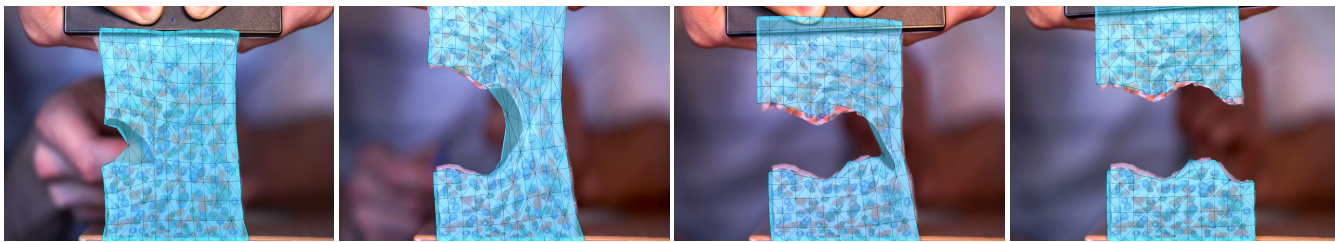


Figure 6: Selected frames from the s-example. The silicone band undergoes both cutting and tearing while being elongated.

and 605 tetrahedral elements.

Figure 4 illustrates the workflow of the proposed method on this example: From the input video stream we extract features f_i and track them over time - for the v-example, we extracted 531 features (figure 4 (left)). We deform the volumetric object by connecting the real features f_i to the virtual features f_i^* that are lying on the objects surface (figure 4 (left middle)). Using the neighborhood information N of the virtual features, the measure introduced in section 4 can be calculated (figure 4 (right middle)). Outliers compared to the average of the measure, define a cut region in which we put the cut points, that are illustrated in figure 4 (right middle) as well. At every update, a cut line between an old and a new cut point is extended to a cut plane, which then introduces the topological change in the volumetric model (figure 4 (right)).

The v-example uses 6795 neighboring virtual features for the detection of an emerging separation. Due to the topological change, the neighborhood information N is updated until we have 6049 neighboring features in the final frame of the video. A topological change such as a cut or a fracture yields more freedom of motion for the object. In this example, the volumetric model makes use of additional 149 vertexes to take account of the appearing separation. Despite the additional computational cost of to the added degrees of freedom the computation remains real-time, above 25 frames per second.

S-Example: The second example considers another cuboid, that is made of a silicone ten times stiffer than the silicone of the first example. Again, the object is deformed by elongation, but this time, we cut horizontally, with a complex cut geometry – similar to a part of the sinusoidal function, the s-example. After the cut is performed, the object is separated into two parts by tearing. Between the three actions – deformation/elongation, the cut and the tearing – the object returns several times to a position with less stress, disturbing the detection of a cut. Selected frames of the augmented reality are displayed in figure 6. The objects dimensions are 8cm in width and height, but the object is thicker 1.2cm. Like in the first example, we apply a virtual model with 288 degrees of freedom and 605 tetrahedral elements. The video stream yields 693 features, that are initially connected by 9225 pairs of neighboring virtual features. In the course of the simulation the number of neighboring virtual features f_i^* reduces to 8023 while the number of vertexes in the volumetric model increases to 315. Similar to the v-example, the impact on the computation time is negligible, the real-time computation is not endangered.

Computational Cost: In both examples the majority of the computational cost is dedicated to solving the minimization problem mentioned in section 3. Our separation detection algorithm only adds a minor computational load – for the v-example less than and for the s-example around – 1% of the complete computational cost, which is equivalent to an average of about 0.20 and 0.25 ms. We consider this as very promising, since the implementation has not been optimized and could be parallelized if necessary. This makes our approach specifically interesting as added functionality for any simulation framework that can deal with topological changes.

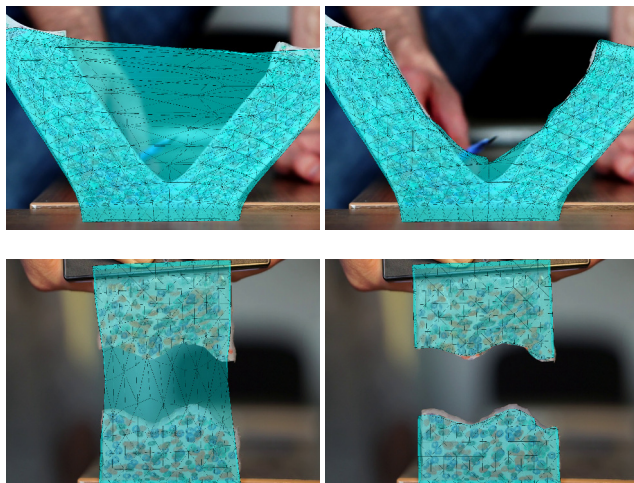


Figure 7: Comparison of an augmentation without and with our method (top) the v-example, (bottom) the s-example

Accuracy: In order to evaluate the accuracy of the method, we use the dice measure comparing the area of the intersection of two sets with the area of the sets themselves. Identically sets yield a dice measure of 1.0, whereas sets that do not intersect have a measure of 0.0. Further information on the calculation and the usage of the dice measure can be found in [11].

Our examples yield very good results for the dice measure, in particular when compared to an augmented reality that would continue using a volumetric model not accounting for topological changes. Visual evaluation can be performed by looking at figure 7. In the first test case – the vertical cut from the top – the dice measure in the final frame using our method is **0.951**, whereas an approach based on an uncut model only has a dice measure of **0.740**. The second example delivers similar results: our method has a dice measure of **0.964** in contrary to an uncut model that can only react incorrectly to the topological change yielding a dice measure of **0.819**.

6 CONCLUSION

In this paper we address the occurrence of topological changes during augmented reality of non-rigid structures. After extracting visual features from a video stream, our method uses these features to control the motion of a virtual representation of a volumetric object. This is done by applying external forces between the feature location and its projection onto the virtual model obtained at time $t = 0$ of the simulation. The virtual model follows a non-linear elastic behavior and is simulated using a finite element method. The difference in the motion between the real and the virtual features is used

to identify discontinuities in the displacement field that are a characteristic of occurring topological changes. When detected, these changes are efficiently incorporated in the physically-based model using a combination of a remeshing algorithm with the snapping of the nodes to the separation. Experiments on real data involving highly elastic silicone bands show the effectiveness of our method when undergoing both cutting and tearing, in the presence of visual tracking disturbance. The experiments highlight the potential of our method, and its possible application to medical augmented reality.

Our method permits to obtain a realistic augmentation of an elastic object even after undergoing a cut or tearing, and this in real-time.

Based on these contributions it would be very interesting to investigate into a partial cutting and tearing of a volume in the direction of the camera. Moreover the intersection of several cuts and fractures is a challenge that will be addressed. Currently, our work relies on monocular images and applications for which only two-dimensional surface tracking is required. Our future work will focus on the integration of a more advanced acquisition system such as depth camera, where 3D deformable objects can be tracked easier.

ACKNOWLEDGMENTS

The authors would like to thank Bruno Marques for his assistance with the image acquisition.

REFERENCES

- [1] A. Agudo, B. Calvo, and J. M. M. Montiel. Finite element based sequential bayesian non-rigid structure from motion. In *CVPR*, pages 1418–1425, 2012.
- [2] A. Bartoli and A. Zisserman. Direct estimation of non-rigid registration. In *British Machine Vision Conference, BMVC 2004, Kingston, UK, September 7-9, 2004. Proceedings*, pages 1–10, 2004.
- [3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.
- [4] J. Y. Bouguet. Pyramidal Implementation of the Lucas Kanade Feature Tracker: Description of the algorithm, 2002.
- [5] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin. SOFA: A Multi-Model Framework for Interactive Physical Simulation. In Y. Payan, editor, *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*. Springer, June 2012.
- [6] S. F. Gibson and B. Mirtich. A survey of deformable modeling in computer graphics. Technical report, Mitsubishi Electronic Research Laboratory, 1997.
- [7] N. Haouchine, J. Dequidt, M.-O. Berger, and S. Cotin. Single View Augmentation of 3D Elastic Objects. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 199–208, 2014.
- [8] N. Haouchine, J. Dequidt, E. Kerrien, M.-O. Berger, and S. Cotin. Physics-based Augmented Reality for 3D Deformable Object. In *VRIPHYS - Virtual Reality Interaction and Physical Simulation*, pages 31–38, Darmstadt, Germany, Dec. 2012.
- [9] N. Haouchine, J. Dequidt, I. Peterlik, E. Kerrien, M.-O. Berger, and S. Cotin. Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 199–208, 2013.
- [10] L. Ibaí, A. Hugo, A. Iker, and B. Diego. Real-time deformation, registration and tracking of solids based on physical simulation. In *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*, pages 165–170, 2014.
- [11] D. Lin. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, 1998.
- [12] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2, IJCAI '81*, pages 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.
- [13] A. Malti, R. Hartley, A. Bartoli, and J.-H. Kim. Monocular template-based 3d reconstruction of extensible surfaces with local linear elasticity. In *Computer Vision and Pattern Recognition, 2013 IEEE Conference on*, pages 1522–1529, 2013.
- [14] F. Moreno-Noguer, M. Salzmann, V. Lepetit, and P. Fua. Capturing 3d stretchable surfaces from single images in closed form. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1842–1849, June 2009.
- [15] A. Nealen, M. Müller, R. Keiser, E. Boxerman, and M. Carlson. Physically based deformable models in computer graphics. In *Computer Graphics Forum*, volume 25 (4), pages 809–836, 2006.
- [16] S. Nicolau, L. Soler, D. Mutter, and J. Marescaux. Augmented reality in laparoscopic surgical oncology. *Surgical Oncology*, 20(3):189 – 201, 2011.
- [17] H.-W. Nienhuys and A. Frank van der Stappen. A surgery simulation supporting cuts and finite element deformation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 2208, pages 145–152. Springer, 2001.
- [18] C. J. Paulus, L. Untereiner, H. Courtecuisse, S. Cotin, and D. Cazier. Virtual cutting of deformable objects based on efficient topological operations. *The Visual Computer*, 31(6-8):831–841, 2015.
- [19] M. Perriollat, R. Hartley, and A. Bartoli. Monocular template-based reconstruction of inextensible surfaces. *International Journal of Computer Vision*, 95(2):124–137, 2011.
- [20] J. Pilet, V. Lepetit, and P. Fua. Fast non-rigid surface detection, registration and realistic augmentation. *Int. J. Comput. Vision*, 76(2):109–122, Feb. 2008.
- [21] P. Pratt, D. Stoyanov, M. Visentini-Scarzanella, and G.-Z. Yang. Dynamic guidance for robotic surgery using image-constrained biomechanical models. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 6361, pages 77–85. Springer, 2010.
- [22] M. Salzmann and P. Fua. Linear local models for monocular reconstruction of deformable surfaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):931–944, May 2011.
- [23] M. Salzmann, R. Hartley, and P. Fua. Convex optimization for deformable surface 3-d tracking. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, Oct 2007.
- [24] M. Salzmann, J. Pilet, S. Ilic, and P. Fua. Surface deformation models for nonrigid 3d shape recovery. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(8):1481–1487, Aug. 2007.
- [25] S. Shen, W. Shi, and Y. Liu. Monocular 3-d tracking of inextensible deformable surfaces under l2-norm. *Trans. Img. Proc.*, 19(2):512–521, Feb. 2010.
- [26] E. Sifakis and J. Barbic. Fem simulation of 3d deformable solids: A practitioner’s guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses, SIGGRAPH '12*, pages 20:1–20:50, New York, NY, USA, 2012. ACM.
- [27] S. Suwelack, S. Röhl, S. Bodenstedt, D. Reichard, R. Dillmann, T. dos Santos, L. Maier-Hein, M. Wagner, J. Wünscher, H. Kennigott, et al. Physics-based shape matching for intraoperative image guidance. *Medical physics*, 41(11):111901, 2014.
- [28] J. Wu, R. Westermann, and C. Dick. Physically-based simulation of cuts in deformable bodies: A survey. In *Eurographics 2014 State-of-the-Art Report*, pages 1–19, Strasbourg, France, 2014. Eurographics Association.
- [29] J. Wu, R. Westermann, and C. Dick. Real-time haptic cutting of high resolution soft tissues. *Studies in Health Technology and Informatics (Proc. Medicine Meets Virtual Reality 2014)*, 196:469–475, 2014. Published by IOS Press.
- [30] S. Wuhrer, J. Lang, and C. Shu. Tracking complete deformable objects with finite elements. In *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*, pages 1–8. IEEE, 2012.
- [31] J. Zhu and M. R. Lyu. Progressive finite newton approach to real-time nonrigid surface detection. In *Computer Vision and Pattern Recognition, 2007. IEEE Conference on*, pages 1–8. IEEE, 2007.