Energy-Based Reconstruction of 3D Curves for Quality Control

H. Martinsson[1], F. Gaspard[1], A. Bartoli[2], and J.-M. Lavest[2]

[1] CEA, LIST, Boîte Courrier 94, F-91 191 Gif sur Yvette, France;
tel: +33(0)1 69 08 82 98, fax: +33(0)1 69 08 83 95
`hanna.martinsson@cea.fr`
[2] LASMEA (CNRS/UBP), 24 avenue des Landais, F-63 177 Aubière, France;
tel: +33(0)4 73 40 76 61, fax: +33(0)4 73 40 72 62
`adrien.bartoli@gmail.fr`

Abstract. In the area of quality control by vision, the reconstruction of 3D curves is a convenient tool to detect and quantify possible anomalies. Whereas other methods exist that allow us to describe surface elements, the contour approach will prove to be useful to reconstruct the object close to discontinuities, such as holes or edges.

We present an algorithm for the reconstruction of 3D parametric curves, based on a fixed complexity model, embedded in an iterative framework of control point insertion. The successive increase of degrees of freedom provides for a good precision while avoiding to over-parameterize the model. The curve is reconstructed by adapting the projections of a 3D NURBS snake to the observed curves in a multi-view setting. The optimization of the curve is performed with respect to the control points using an gradient-based energy minimization method, whereas the insertion procedure relies on the computation of the distance from the curve to the image edges.

1 Introduction

The use of optical sensors in metrology applications is a complicated task when dealing with complex or irregular structures. More precisely, projection of structured light allows for an accurate reconstruction of surface points but does not allow for a precise localization of the discontinuities of the object. This paper deals with the problem of reconstruction of 3D curves, given the CAD model, for the purpose of a control of conformity with respect to this model. We dispose of a set of images with given perspective projection matrices. The reconstruction will be accomplished by means of the observed contours and their matching, both across the images and to the model. We proposed a previous version of our algorithm, based on edge distances, in [12]. The contributions of this paper with respect to the former one resides in the energy formulation, giving a new structure to the problem. We have also completed the experimental evaluation.

Algorithms based on active contours [11] allows for a local adjustment of the model and a precise reconstruction of primitives. More precisely, the method allows for an evolution of the reprojected model curves toward the image edges,

thus to minimize the distance in the images between the predicted curves and the observed edges.

The parameterization of the curves as well as the optimization algorithms we use must yield an estimate that meets the requirements of accuracy and robustness necessary to perform a control of conformity. We have chosen to use NURBS curves [14], a powerful mathematical tool that is also widely used in industrial applications.

In order to ensure stability, any method used ought to be robust to erroneous data, namely the primitives extracted from the images, since images of metallic objects incorporate numerous false edges due to reflections.

Although initially defined for ordered point clouds, active contours have been adapted to parametric curves. Cham and Cipolla propose a method based on affine epipolar geometry [4] that reconstructs a parametric curve in a canonical frame using stereo vision. The result is two coupled snakes, but without directly expressing the 3D points. In [19], Xiao and Li deal with the problem of reconstruction of 3D curves from two images. However, the NURBS curves are approximated by B-splines, which makes the problem linear, at the expense of loosing projective invariance. The reconstruction is based on a matching process using epipolar geometry followed by triangulation. The estimation of the curves is performed independently in the two images, that is, there is no interactivity between the 2D observations and the 3D curve in the optimization. Kahl and August introduce in [10] a coupling between matching and reconstruction, based on an a priori known distribution of the curves and on an image formation model. The curves are expressed as B-splines and the optimization is done using gradient descent.

Other problems related to the estimation of parametric structures have come up in the area of surfaces. In [18], Siddiqui and Sclaroff present a method to reconstruct rational B-spline surfaces. Point correspondences are supposed given. In a first step, B-spline surface patches are estimated in each view, then the surface in 3D, together with the projection matrices, are computed using factorization. Finally, the surface and the projection matrices are refined iteratively by minimizing the 2D residual error. So as to avoid problems due to over-parameterization, the number of control points is limited initially, to be increased later on in a hierarchical process by control point insertion.

In the field of medical imaging, energy minimization methods have been developed to reconstruct 3D curves in a stereo setting. Sbert and Solé reconstruct in [16] a 3D curve using an energy based evolution method. The associated PDE of the energy functional, derived by the Euler-Lagrange formulation, is solved using a level-set approach. In [3], Canero et al. define in a force field by reprojecting external image forces, given by the distance to the edges. A 3D curve is then reconstructed via the evolution of an active contour, guided by the force field.

In the case of 2D curve estimation, other aspects of the problem are addressed. Cham and Cipolla adjust a spline curve to fit an image contour [5]. Control points are inserted iteratively using a new method called PERM (poten-

tial for energy-reduction maximization). An MDL (minimal description length [9]) strategy is used to define a stopping criterion. In order to update the curve, the actual curve is sampled and a line-search is performed in the image to localize the target shape. The optimization is performed by gradient descent. Brigger et al. present in [2] a B-spline snake method without internal energy, due to the intrinsic regularity of B-spline curves. The optimization is done on the knot points rather than on the control points, which allows the formulation of a system of equations that can be solved by digital filtering. So as to increase numerical stability, the method is embedded in a multi-resolution framework. In [8], Figueiredo et al. address the problem from a statistical point of view, proposing a completely automatic contour estimator, in the sense that no parameter need to be adjusted by the user. Supposing a uniform distribution of the knot points, the B-spline curve that approximates a given set of contour points at best, in the least squares sense, is given by a linear system depending only on the number of control points. This number is fixed in advance using an MDL criterion. Meegama and Rajapakse introduce in [13] an adaptive procedure for control point insertion and deletion, based on the euclidean distance between consecutive control points and on the curvature of the NURBS curve. Local control is ensured by adjustment of the weights. The control points evolve in each iteration in a small neighborhood (3×3 pixels). Yang et al. use a distance field computed a priori with the fast marching method in order to adjust a B-spline snake [20]. Control points are added in the segment presenting a large estimation error, due to a degree of freedom insufficient for a good fit of the curve. The procedure is repeated until the error is lower than a fixed threshold. Redundant control points are then removed, as long as the error remains lower than the threshold.

2 Problem Formulation

Given a set of images of an object, together with its CAD model, our goal is to reconstruct in 3D the curves observed in the images. The reconstruction is performed by minimizing an energy functional. In order to obtain a 3D curve that meets our requirements regarding regularity, rather than reconstructing a point cloud, we estimate a NURBS curve. Since the regularity aspects are thereby taken care of, the energy functional is defined solely based on image data. The minimization problem is formulated for a set of M images and N sample points by

$$\mathbf{C}(\mathcal{P}) = \arg\min_{\mathcal{P}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} E(T_i(\mathbf{C}(\mathcal{P}, t_j))), \tag{1}$$

where E is the external energy functional, T_i is the projective operator for image i and $\mathcal{P}$ is the set of control points.

Our choice to use NURBS curves is justified by several reasons. First, NURBS curves have interesting geometrical properties, namely concerning regularity and continuity. An important geometrical property that will be of particular interest is the invariance under projective transformations.

3 Properties of NURBS curves

Let $U = \{u_0, \cdots, u_m\}$ be an increasing vector, called the knot vector. A NURBS curve is a vector valued, piecewise rational polynomial over U, defined by

$$\mathbf{C}(t) = \sum_{i=0}^{n} \mathbf{P}_i R_{i,k}(t) \quad \text{with} \quad R_{i,k}(t) = \frac{w_i B_{i,k}(t)}{\sum_{j=0}^{n} w_j B_{j,k}(t)}, \tag{2}$$

where $\mathbf{P}_i$ are the control points, $B_{i,k}(t)$ the B-spline basis functions defined over U, w_i the associated weights and k the degree.

It is a common choice to take $k = 3$, which has proved to be a good compromise between required smoothness and the problem of oscillation, inherent to high degree polynomials. For our purposes, the parameterization of closed curves, we consider periodic knot vectors, that is, verifying $u_{j+m} = u_j$. Given all these parameters, the set of NURBS defined on U forms, together with the operations of point-wise addition and multiplication with a scalar, a vector space.

For details on NURBS curves and their properties, refer to [14].

3.1 Projective Invariance

According to the pinhole camera model, the perspective projection $T(\cdot)$ that transforms a world point into an image point is expressed in homogeneous coordinates by means of the transformation matrix $\mathbf{T}_{3\times 4}$. Using weights associated with the control points, NURBS curves have the important property of being invariant under projective transformations. Indeed, the projection of (2) remains a NURBS, defined by its projected control points and their modified weights. The curve is written

$$\mathbf{c}(t) = T(\mathbf{C})(t) = \frac{\sum_{i=0}^{n} w_i' \, T(\mathbf{P}_i) \, B_{i,k}(t)}{\sum_{i=0}^{n} w_i' \, B_{i,k}(t)} = \sum_{i=0}^{n} T(\mathbf{P}_i) R_{i,k}'(t), \tag{3}$$

where the $R_{i,k}'$ are the basis functions of the projected NURBS. The new weights, w_i', are given by

$$w_i' = (T_{3,1} X_i + T_{3,2} Y_i + T_{3,3} Z_i + T_{3,4}) \, w_i = \mathbf{n} \cdot (\mathbf{C}_\mathbf{O} - \mathbf{P}_i) \, w_i, \tag{4}$$

where $\mathbf{n}$ is a unit vector along the optical axis and $\mathbf{C}_\mathbf{O}$ the optical center of the camera.

3.2 Control Point Insertion

One of the fundamental geometric algorithms available for NURBS curves is the control point insertion. The key is the knot insertion, which is equivalent to adding one dimension to the vector space, consequently adapting the basis. Since the original vector space is included in the new one, there is a set of control points such that the curve remains unchanged.

Let $\bar{u} \in [u_j, u_{j+1})$. We insert $\bar{u}$ in U, forming the new knot vector $\bar{U} = \{\bar{u}_0 = u_0, \cdots, \bar{u}_j = u_j, \bar{u}_{j+1} = \bar{u}, \bar{u}_{j+2} = u_{j+1}, \cdots, \bar{u}_{m+1} = u_m\}$. The new control points $\bar{\mathbf{P}}_i$ are given by the linear system

$$\sum_{i=0}^{n} \mathbf{P}_i R_{i,k}(t) = \sum_{i=0}^{n+1} \bar{\mathbf{P}}_i \bar{R}_{i,k}(t).$$

(5)

We present the solution without proof. The new control points are written [14]

$$\bar{\mathbf{P}}_i = \alpha_i \, \mathbf{P}_i + (1 - \alpha_i) \, \mathbf{P}_{i-1},$$

(6)

with

$$\alpha_i = \begin{cases} 1 & i \le j - k \\ \dfrac{\bar{u} - u_i}{u_{i+k} - u_i} & \text{if } \ j - k + 1 \le i \le j \\ 0 & i \ge j + 1 \end{cases} .$$

(7)

Note that only k new control points need to be computed, due to the local influence of splines.

4   Optimization

When treating NURBS curves, the regularity aspects are taken care of implicitly by the parameterization and the energy functional can be reduced to its external energy part. We will consider two forms of energy functionals, one based on the distance from the curve to the image contours and another one based on the gradient intensity. The optimization will in both cases operate on the control points of the 3D NURBS curve.

4.1   Distance Minimization

Using a distance formulation and the properties of NURBS curves, the minimization problem (1) is written

$$\min_{\{\hat{\mathbf{P}}_l\}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left(\mathbf{q}_{ij} - \sum_{l=0}^{n} T_i(\hat{\mathbf{P}}_l) R_{l,k}^{(i)}(t_j) \right)^2 ,$$

(8)

where $\mathbf{q}_{ij}$ is a contour point associated with the curve point of parameter t_j in image i and T_i is the projective operator for image i. The search for candidate contour points is carried out independently in the images using a method inspired by the one used by Drummond and Cipolla in [7].

Search for Image Contours We sample the NURBS curve projected in the image, to use as starting points in the search for matching contour points. A line-search is performed in order to find the new position of the curve, ideally corresponding to an edge. Our approach is based solely on the contours. Due

to the aperture problem, the component of motion of an edge, tangent to itself, is not detectable locally and we therefore restrict the search for the new edge position to the edge normal at each sample point. As we expect the motion to be small, we define a search range (typically in the order of 20 pixels) so as to limit computational cost. In order to find the new position of a sample point, for each point belonging to the normal within the range, we evaluate the gradient and compute a weight based on the intensity and the orientation of the gradient and the distance from the sample point. The weight function v_j for a sample point p_j and the candidate point p_ξ will be of the form

$$v_j(p_\xi) = \varphi_1(|\nabla I_\xi|) \cdot \varphi_2\left(\frac{\hat{n}_j \cdot \nabla I_\xi}{|\nabla I_\xi|}\right) \cdot \varphi_3(|p_j - p_\xi|),$$

where $\hat{n}_j$ is the normal of the projected curve at sample point j, ∇I_ξ is the gradient at the candidate point and the φ_k are functions to define. The weight function will be evaluated for each candidate p_ξ and the point p'_j with the highest weight, identified by its distance from the original point $d_j = |p_j - p'_j|$, will be retained as the candidate for the new position of the point.

The bounded search range and the weighting of the point based on their distance from the curve yield a robust behavior, close to that of an M-estimator.

4.2 Gradient Energy Minimization

Using the classical energy formulation and the properties of NURBS curves, the minimization problem (1) is written

$$\min_{\{\hat{\mathbf{P}}_l\}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} E\left(\sum_{l=0}^{n} T_i(\hat{\mathbf{P}}_l) R_{l,k}^{(i)}(t_j)\right), \tag{9}$$

where $R_{l,k}^{(i)}$ are the basis functions for the projected NURBS curve in image i. The energy functional E can, as already mentioned, be restricted to its external part, due to the use of NURBS. A common choice is to use the gradient intensity. We will however include local information on the curve, namely its normal direction, using the intensity of the gradient projected onto the curve normal.

4.3 Distance versus Gradient Energy

For comparison, we have implemented the two methods in the iterative setting that will be introduced in the following section. Both methods yielded similar results and converge after a number of iterations to an asymptotic lower limit. The 3D error with respect to the true curve is however somewhat lower for the gradient-based method. The results are given in Fig. 2. The difference is partly explained by the noise and the parallel structures perturbing the edge tracking algorithm. An example of candidate points located on a parallel image contour, due to specularities, is given in Fig. 1. Although the gradient intensity method outperforms the distance method, the distance-based cost function will prove to be useful in the iterative framework that will embed the curve optimization.
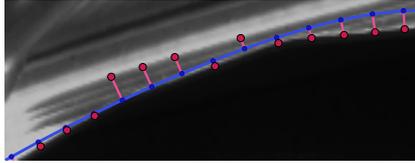
Fig. 1. Problems related to specularities and to the search for candidate points. Starting at the projection of the initial curve (in blue), some candidate points (in magenta) belong to a parasite edge.
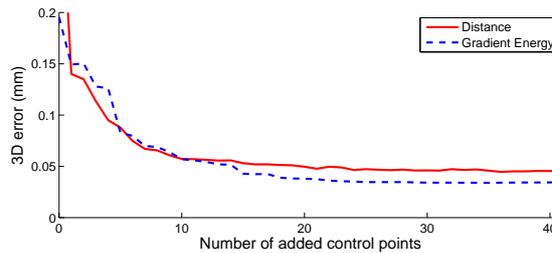

Fig. 2. The evolution of the error, with respect to the true 3D curve, for an optimization using cost functions based on the distance to the image contours and on the gradient intensity respectively. Whereas the distance method seems to yield good results in the start, its asymptotic limit is somewhat higher than that of the gradient intensity method.

5 Curve Estimation

The problem has two parts. First, the optimization of the 3D NURBS curve by energy minimization on a fixed number of control points, then the control point insertion procedure. For the fixed size optimization problem, we use the non-linear Levenberg-Marquardt minimization method. This step allows the control points to move in 3D, but does not change their number. In order to obtain an optimal reconstruction of the observed curve, we iteratively perform control point insertion. So as to avoid over-parameterization for stability reasons, the first optimization is carried out on a limited number of control points. Their number is then increased by iterative insertion, so that the estimated 3D curve fits correctly also in high curvature regions. As mentioned earlier, the insertion of a control point is done without influence on the curve and a second optimization is thus necessary in order to take advantage from the increased number of degrees of freedom.

5.1 Optimization on the Control Points

The first step of the optimization consists in projecting the curve in the images. Since the surface model is known, we can identify the visible parts of the curve
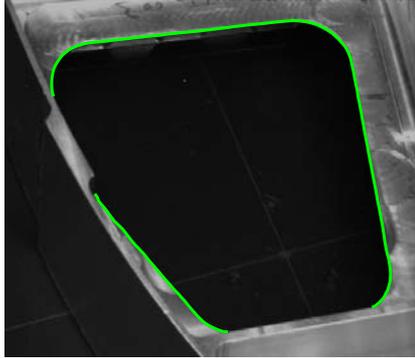
Fig. 3. Due to the use of 3D objects, auto-occlusions cause parts of the curve to be invisible from some viewpoints.

in each image and retain only the sample points corresponding to visible parts. During the iterations, to keep the same cost function, the residual error must be evaluated in the same points in each iteration. Supposing small displacements, we can consider that visible pieces will remain visible throughout the optimization. See Fig. 3 for an example of occlusions due to the 3D structure of the objects.

The optimization of (9) is done on the 3D control point coordinates, leaving the remaining parameters of the NURBS curve constant. The weights associated with the control points are modified by the projection giving 2D weights varying with the depth of each control point, according to the formula (4), but they are not subject to the optimization.

5.2 Control Point Insertion

Due to the use of NURBS, we have a method to insert control points. What remains is to decide where to place them. We also need a criterion to decide when to stop the control point insertion procedure.

Position of the New Control Point Several strategies have been used. Cham and Cipolla consider in [5] the dual problem of knot insertion. They define an error energy reduction potential and propose to place the knot point so as to maximize this potential. The control point is placed using the method described earlier. In our algorithm, since every insertion is followed by an optimization that adjusts the control points, we settle for choosing the interval where to place the point. Since the exact location within the interval is not critical, the point is placed at its midpoint. Dierckx suggests in [6] to place the new point at the interval that presents the highest error. This is consistent with an interpretation of the error as the result of a lack of degrees of freedom that inhibits a good description of the curve. If, however, the error derives from other sources, this solution is not always optimal.

In our case, a significant mean error could also indicate the presence of parasite edges or that of a parallel structure close to the target curve. We will therefore choose the interval with the highest median error, over all images. The error is defined as the distance from a sample point to its corresponding contour point in the image. The search for candidate contour points is carried out using the method described in 4.1.

Stopping Criterion One of the motives for introducing parametric curves was to avoid treating all curve points, as only the control points are modified during the optimization. If the number of control points is close to the number of samples, the benefit is limited. Too many control points could also cause numerical instabilities, due to an over-parameterization of the curve on the one hand and the size of the non-linear minimization problem on the other hand. It is thus necessary to define a criterion that decides when to stop the control point insertion.

A strategy that aims to avoid the over-parameterization is the use of statistical methods inspired by the information theory. Based in a Maximum Likelihood environment, these methods combine a term equivalent, in the case of a normal distributed errors, to the sum of squares of the residual errors with a term penalizing the model complexity. Given two estimated models, in our case differentiated by their number of control points, the one with the lowest criterion will be retained. The first criterion of this type, called AIC (Akaike Information Criterion), was introduced by Akaike in [1] and is written, in the case of normally distributed errors,

$$AIC = 2k + n \ln \frac{RSS}{n},$$ (10)

where k is the number of control points, n is the number of observations and RSS is the sum of the squared residual errors. Another criterion, based on a bayesian formalism, is the BIC (Bayesian Information Criterion) presented by Schwarz [17]. It stresses the number of data points n, so as to ensure an asymptotic consistency and is written, also in the case of normally distributed errors,

$$BIC = 2k \ln n + n \ln \frac{RSS}{n}.$$ (11)

Another family of methods uses the MDL [15] formulation, which consists in associating a cost with the quantity of information necessary to describe the curve. Different criteria follow, depending on the formulation of the estimation problem. In the iterative control point insertion procedure of Cham et Cipolla [5], the stopping criterion is defined by means of MDL. The criterion depends, on the one hand on the number of control points and on the residual errors, on the other hand on the number of samples and on the covariance.

Yet another way of choosing an appropriate model complexity is the classical method of cross-validation. The models are evaluated based on their capacity to describe the data. A subset of the data is used to define a fixed complexity

model, while the rest serve to validate it. The process is repeated and a model is retained if its performance is considered good enough.

We have chosen to use the BIC, computed using the contour points found with the method presented in 4.1, for this first version of our algorithm. A more thorough study of the influence of the stopping criterion in our setting will be performed at a later stage.

5.3 Algorithm

The algorithm we implemented has two layers. The optimization of a curve using a fixed complexity model is embedded in an iterative structure that aims to increase the number of control points. The non-linear optimization of the 3D curve is performed by the Levenberg-Marquardt algorithm, using a cost function based on an energy formulation. The control point insertion procedure uses a search for contour points in the images in order to compute the median as well as the RSS error of the projected curve. The mechanism of our method is outlined in Table 1.

6 Experimental Evaluation

6.1 Virtual Images

In order to validate our algorithms for image data extraction and for curve reconstruction, we have performed a number of tests on virtual images. The virtual setting also allows us to simulate deformations of the target object.

We construct a simplified model of an object, based on a single target curve. We then apply our 3D reconstruction algorithm, starting at a modified "model curve", on a set of virtual views, see Fig. 4. The image size is 1284×1002 pixels. The starting curve has 10 control points, to which 45 new points are added. The sampling used for the computations is of 200 points. To fix the scale, note that at the mean distance from the object curve, one pixel corresponds roughly to 0.22 mm. The evaluation of the results is done by measuring the distance from a set of sampled points from the estimated curve to the target model curve. The distances from the target curve are shown in Fig. 4. We obtain the following results:

Mean error	0.0336 mm
Median error	0.0228 mm
Standard deviation	0.0485 mm

We note that the error corresponds to less than a pixel in the images, which indicates a sub-pixel image precision.

Using a series of virtual images of an object presenting a minor anomaly, we have also tested the capacity of our system to detect nonconformities, see Fig. 5. Based on 27 images and starting at the model curve, our algorithm manages to reconstruct the curve and its anomaly with a mean error of 0.0765 mm. Although the reconstruction is good, the error is concentrated around the anomaly, which is somewhat smoothed out.
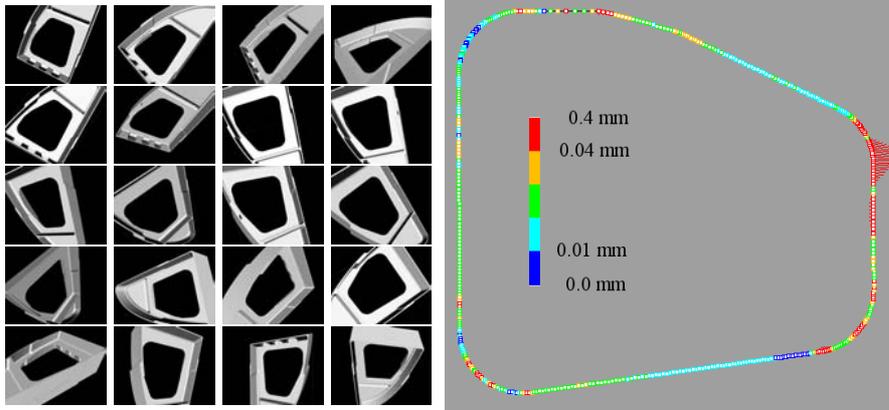
Fig. 4. *Left:* Some of the 32 virtual images used for the reconstruction of the central curve. *Right:* The distances from the sampled points from the reconstructed 3D curve to the model curve. The cloud of sample points from the estimated curve is shown together with the target curve. The starting curve is shown in black. The differences are represented by lines with length proportional to the distance between the curve and the target, using a scale factor of 20.

6.2 Real Images

We also consider a set of real images, see Fig. 6, with the same target curve, using the same starting "model curve" as in the virtual case. We now need to face the problem of noisy image data, multiple parallel structures and imprecision in the localization and the calibration of the views. The image size is 1392×1040 pixels. The starting curve has 10 control points, to which 48 new points are added. The sampling used for the computations is of 200 points. At the mean distance from the object curve, one pixel corresponds roughly to 0.28 mm. The distances from the target curve are shown in Fig. 6. We obtain the following results:

Mean error	0.137 mm
Median error	0.125 mm
Standard deviation	0.074 mm

Even if the errors are higher than in the case of virtual images, we note that they still correspond to less than a pixel in the images. The difference is explained by the noise and to some extent by specularities, causing parallel structures perturbing the minimization algorithm, see Fig. 7(b).

The evolution of the control points is demonstrated in Fig. 7(a), where the set of initial control points is shown, together with the final curve and its control points. As expected, the control points inserted are concentrated in the regions of high curvature, such as the corners.
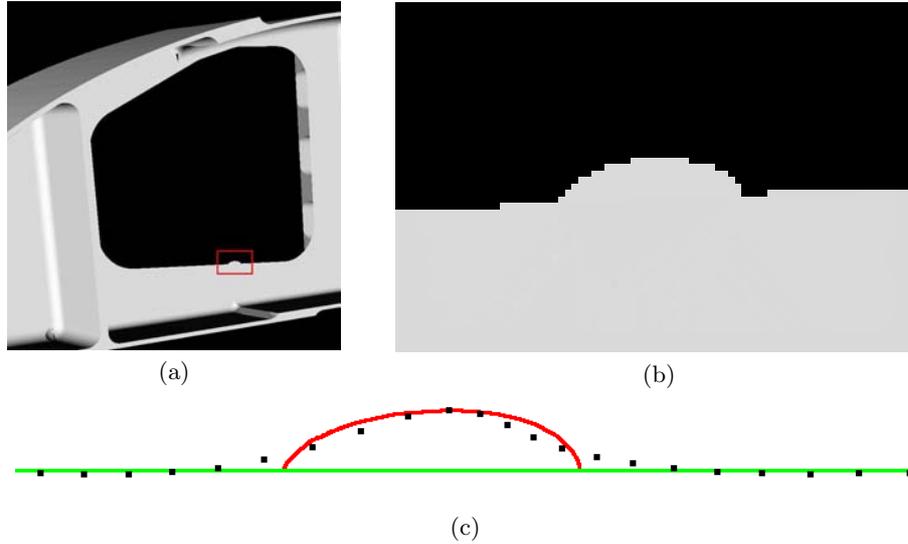
(a)   (b)

(c)

Fig. 5. Reconstruction of a nonconformity based on a series of virtual images of an object with an anomaly. The object is shown in (a) with the anomaly marked in red, with a close-up in (b). The result of the reconstruction around the anomaly is shown in (c), with the original curve in green, the anomaly in red and the reconstructed points in black.

7 Conclusions

We have presented an adaptive 3D reconstruction method using parametric curves, limiting the degrees of freedom of the problem. An algorithm for 3D reconstruction of curves using a fixed complexity model is embedded in an iterative framework, allowing an enhanced approximation by control point insertion. The optimization of the curve with respect to the control points is performed by means of a minimization of an gradient-based energy functional, whereas the insertion procedure is based on the distance from the curve to the observed image contours. An experimental evaluation of the method, using virtual as well as real images, has let us validate its performance in some simple, nevertheless realistic, cases with specular objects subject to occlusions and noise.

Future work will be devoted to the integration of knowledge of the CAD model in the image based edge tracking. Considering the expected neighborhood of a sample point, the problem of parasite contours should be controlled and has limited impact on the obtained precision.

We also plan to do a deeper study around the stopping criterion used in the control point insertion process, using cross-validation.
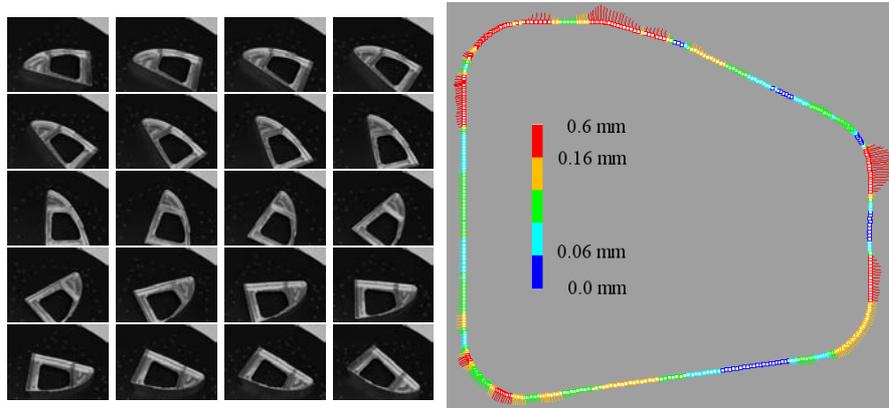
Fig. 6. *Left:* Some of the 36 real images used for the reconstruction of the curve describing the central hole. *Right:* The distances from the sampled points from the reconstructed 3D curve to the model curve. The differences are represented by lines with length proportional to the distance between the curve and the target, using a scale factor of 20.

References

1. H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automated Control*, 19(6):716–723, 1974.
2. P. Brigger, J. Hoeg, and M. Unser. B-spline snakes: A flexible tool for parametric contour detection. *IEEE Trans. on Image Processing*, 9(9):1484–1496, July 2000.
3. C. Canero, P. Radeva, R. Toledo, J.J. Villanueva, and J. Mauri. 3D curve reconstruction by biplane snakes. In *15th International Conference on Pattern Recognition (ICPR'00)*, volume 4, pages 563–566, 2000.
4. T.-J. Cham and R. Cipolla. Stereo coupled active contours. In *Conference on Computer Vision and Pattern Recognition*, pages 1094–1099. IEEE Computer Society, 1997.
5. T.-J. Cham and R. Cipolla. Automated B-spline curve representation incorporating MDL and error-minimizing control point insertion strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(1):49–53, January 1999.
6. P. Dierckx. *Curve and Surface Fitting with Splines*. Oxford University Press, Inc., New York, NY, USA, 1993.
7. T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:932–946, 2002.
8. M. Figueiredo, J. Leitão, and A.K. Jain. Unsupervised contour representation and estimation using B-splines and a minimum description length criterion. *IEEE Transactions on Image Processing*, 9(6):1075–1087, June 2000.
9. M. H. Hansen and B. Yu. Model selection and the principle of minimum description length. *Journal of the American Statistical Association*, 96(454):746–774, 2001.
10. F. Kahl and J. August. Multiview reconstruction of space curves. In *9th International Conference on Computer Vision*, volume 2, pages 1017–1024, 2003.
11. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 4(1):321–331, 1987.
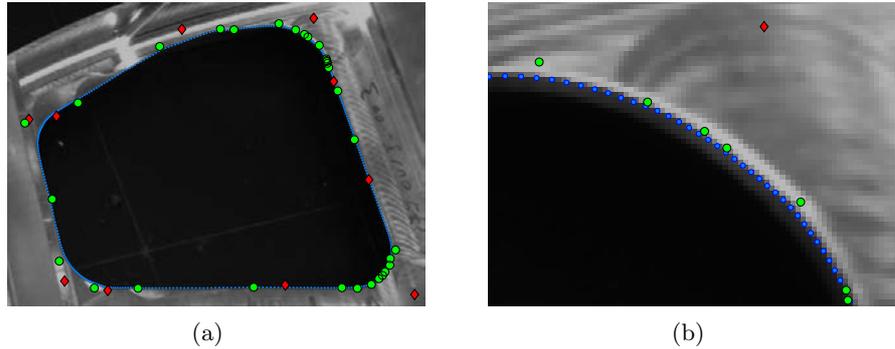
(a)   (b)

Fig. 7. (a) Evolution of the control points. In red the 10 initial control points, in green the 30 control points after optimization and in blue the final curve. (b) A detail, showing the curve between two similar contours, the parallel model curve to the lower left and a false edge caused by specularities to the upper right. Thanks to the multi-view setting, the reconstructed curve corresponds to the true 3D curve.

12. H. Martinsson, F. Gaspard, A. Bartoli, and J-M. Lavest. Reconstruction of 3d curves for quality control. In *15th Scandinavian Conference on Image Analysis*, 2007 (to appear).
13. R.G.N. Meegama and J. C. Rajapakse. NURBS snakes. *Image and Vision Computing*, 21:551–562, 2003.
14. L. Piegl and W. Tiller. *The NURBS book*. Monographs in visual communication. Springer Verlag, 2nd edition, 1997.
15. J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.
16. C. Sbert and A.F. Solé. Stereo reconstruction of 3d curves. In *15th International Conference on Pattern Recognition (ICPR'00)*, volume 1, 2000.
17. G. Schwarz. Estimating the dimension of a model. *Ann. of Stat.*, 6:461–464, 1978.
18. M. Siddiqui and S. Sclaroff. Surface reconstruction from multiple views using rational B-splines and knot insertion. In *First International Symposium on 3D Data Processing Visualization and Transmission*, pages 372–378, 2002.
19. Y.J. Xiao and Y.F. Li. Stereo vision based on perspective invariance of NURBS curves. In *IEEE International Conference on Mechatronics and Machine Vision in Practice*, volume 2, pages 51–56, 2001.
20. H. Yang, W. Wang, and J. Sun. Control point adjustment for B-spline curve approximation. *Computer-Aided Design*, 36:639–652, 2004.

OBJECTIVE

Given a 3D NURBS curve extracted from the CAD model, (partially) seen in M images, sampled in N points. We want to reconstruct the 3D curve observed in the images in order to compare it to the model.

ALGORITHM

- **Visibility Check** Identification of the visible parts

$$\chi_{ij} = \begin{cases} 1 \text{ if } \sum_{l=0}^{n-1} T_i(\mathbf{P}_l)\mathbf{R}_{l,k}^{(i)}(t_j) \text{ visible} \\ 0 \text{ sinon} \end{cases}$$

- **Optimization** on the control points

$$\min_{\{\hat{\mathbf{P}}_l\}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \chi_{ij} \, E\left(\sum_{l=0}^{n-1} T_i(\hat{\mathbf{P}}_l)\mathbf{R}_{l,k}^{(i)}(t_j) \right)$$

- **Line-search** for contour points $\mathbf{q}_{ij}$ matching $\mathbf{p}_{ij} = \sum_{l=0}^{n-1} T_i(\mathbf{P}_l)\mathbf{R}_{l,k}^{(i)}(t_j)$

$$\mathbf{q}_{ij} = \arg_{\mathbf{p}_{ij}^m} \max_{-d \le m \le d} v_j(\mathbf{p}_{ij}^m,) \text{ where } \mathbf{p}_{ij}^m = \mathbf{p}_{ij} + m \cdot \hat{\mathbf{n}}_{ij}$$

- **Computation of the** BIC

$$BIC_0 = k \ln(N \cdot M) + N \cdot M \cdot \ln\left(\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \chi_{ij} \left(\mathbf{q}_{ij} - \sum_{l=0}^{n-1} T_i(\mathbf{P'}_l)\mathbf{R}_{l,k}^{(i)}(t_j) \right)^2 / (N \cdot M) \right)$$

- **do** (control point insertion)
  - **Line-search** for contour points $\mathbf{q}_{ij}$
  - **Computation of the median error** for each interval I_K.

$$m_K = \operatorname*{med}_{E_K} |\mathbf{q}_{ij} - \sum_{l=0}^{n-1} T_i(\mathbf{P'}_l)\mathbf{R}_{l,k}^{(i)}(t_j)|$$

  where $E_K = \{(i,j) \,|\, 0 \le i < M, t_j \in I_K, \chi_{ij} \ne 0\}$.
  - **Knot point insertion** at the midpoint of interval $I = \arg\min_K m_K$.
  - **Visibility Check** Identification of the visible parts
  - **Optimization** on the control points
  - **Computation of the** BIC_J
- **while** $(BIC_J < BIC_{J-1})$

Table 1. Reconstruction algorithm presented.