An algorithm for automatic surface labeling of planar surgical resections

F E Milano\textsuperscript{1,2,*}, L E Ritacco\textsuperscript{2}, G Farfalli\textsuperscript{3}, L Aponte-Tinao\textsuperscript{3}, F González Bernaldo de Quirós\textsuperscript{2} and M Risk\textsuperscript{1,4}

\textsuperscript{1} Instituto Tecnológico de Buenos Aires, Buenos Aires, Argentina
\textsuperscript{2} Unidad de Planeamiento y Navegación Virtual, Hospital Italiano de Buenos Aires, Buenos Aires, Argentina
\textsuperscript{3} Departamento de Oncología Ortopédica, Hospital Italiano de Buenos Aires, Buenos Aires, Argentina
\textsuperscript{4} Consejo Nacional de Investigaciones Científicas y Técnicas - CONICET, Argentina

E-mail: fmilano@ieee.org

\textbf{Abstract.} Three dimensional (3D) preoperative planning and navigation in bone tumor resections have been used in the last five years with good results. The purpose of this study is to develop a method capable of detecting and labeling the nearly planar surface generated by the cutting saw in the surgical specimen taken off the patient during the resection procedure. This surface area labeling is fundamental to track the path that the cutting saw took during the surgery and compare it to the planned cutting plane. The algorithm presented here works by using a 3D reconstruction of the surgical specimen computed tomography (CT) scan, registered against the 3D reconstruction of the preoperative patient CT scan, and the cutting plane defined during surgical planning. The results show a high labeling accuracy (a matching mean of 98.5\%) and a non significant accuracy variation for a range of distance and angle offsets.

1. Introduction

Three dimensional (3D) preoperative planning and navigation in bone tumor resections have been used in the last five years with good results [1]. These results were evaluated histologically, considering free margin from tumor. However, accuracy of preoperative planning [2] and navigation is not yet clear [3]. The purpose of this study is to develop a method capable of detecting and labeling the nearly planar surface generated by the cutting saw in the surgical specimen taken off the patient during the resection procedure. This surface area labeling is fundamental to track the path that the cutting saw took during the surgery and compare it to the planned cutting plane (PCP). The algorithm presented here works by using a 3D reconstruction [4] of the surgical specimen computed tomography (CT) scan, registered against the 3D reconstruction of the preoperative patient CT scan, and the cutting plane defined during surgical planning.

There have been reported algorithms for planar region detection on 3D laser scans of building facades [5] and for Lidar images roof detection [6]. Both algorithms work on point clouds, not taking advantage of information present in triangular meshes. In this work we present an

\*This work was partially financed by a CONICET-UTN doctoral scholarship.
algorithm that maintains its accuracy even though the planar surface detected is surrounded by complex bone geometries.

2. Materials and Methods

2.1. Experimental Data and Preprocessing

The method presented here was developed, tested, and validated using a set of two patient CT images. Each patient comprises two image acquisitions: (1) The preoperative CT scan and (2) the CT scan of the surgical specimen. In this context the surgical specimen is the piece of bone containing the tumor, resected from the patient during the surgery. Both images are acquired using the same multi-slice CT scanner, applying the same acquisition protocol. Both images are 3D reconstructed using a custom medical image processing platform. The reconstruction is achieved by manually thresholding each tomographic slice to obtain a set of binary images that are then voxelised. This volume is then transformed into a closed surface represented by a triangular mesh. Both the preoperative mesh volume (PMV) and the surgical specimen mesh volume (SMV) are then set in a common frame of reference by a manual landmark registration [7], refined by an iterative closest points registration [8].

Figure 1: A planned surgery, showing two cutting planes in red and a bone tumor in green.

Our main objective is to evaluate the accuracy of the postoperative computed cutting surface (CCS) labeling.

2.2. Pipeline

The overall application of this method can be briefly described as follows: having a patient’s PMV, the surgeons that will intervene in the procedure plan ahead where the bone affected by the tumor will be cut, taking into account medical factors such as the tumoral oncologic margin and the particular approach to the surgery. This plan is reflected in a set of PCPs located by the surgeons on the PMV. The plan together with the PMV is uploaded to a surgical navigator and the procedure is executed under intraoperative surgical guidance. After surgery the surgical specimen is CT scanned and reconstructed, obtaining a SMV. Now both PMV and SMV are registered in a common reference frame. The hereby presented tool is then used with the PCP and the SMV inputs to detect the computed cutting surface (CCS) created by the cutting saw. This way it is possible to compare what was planned in the PCP and what was actually executed
2.3. Planar Resection Region Labeling Algorithm

The algorithm is divided in two phases:

(i) Each point of the SMV is projected onto the PCP.

Let the PCP be described by the plane

\[ ax + by + cz + d = 0 \]

with the normal vector \( \mathbf{n} = (a \ b \ c) \) and \( \mathbf{x} = (x \ y \ z) \).

Let SMV be a triangulated 2-manifold representing the surgical specimen. We look for the set of points \( K \) that belong to the SMV and minimize their normal distance to the PCP. Formally:

\[
K = \left\{ k \mid k, u \in \text{SMV and } (\forall x \in \text{PCP})(k = \arg\min_u \left| \frac{n \cdot (u - x)}{|n|} \right|) \right\} \tag{1}
\]

In practice, as the SMV is represented by a mesh built on an unstructured grid and it has potentially infinite points to project, the computation is performed in a different way. The PCP is used to generate a 2-dimensional uniform sampling grid, applying a homogeneous spacing in both plane’s spanning directions. This grid is bounded by the projection of the bounding box of the SMV onto the PCP. Each point of the sampling space defined by this structured grid is projected onto the SMV in the line along the normal vector of the PCP. The closest point in SMV intersected by the projection of the sampling point is then added to the point set \( K \) that is going to be used in the second part of the algorithm. This is described in algorithm 1.

Algorithm 1 Projection

Input: SMV, PCP

Output: \( K \), a set of points of the SMV that fulfills equation 1

1: \( UG \leftarrow \text{Generate Uniform Grid from PCP} \)
2: \( K \leftarrow \emptyset \)
3: \( n \leftarrow n/\text{norm}(n) \)
4: \( \text{threshold} \leftarrow 20 \)
5: for each \( u \) in \( UG \) do
6: \( \text{min} \_ \text{distance} \leftarrow \text{Largest Possible Number} \)
7: \( \text{distance} \leftarrow 0 \)
8: while \( \text{distance} < \text{threshold} \) do
9: \( \text{distance} \leftarrow \text{distance} + 0.01 \)
10: \( k \leftarrow u + n \cdot \text{distance} \)
11: if \( \text{Intersected}(k, \text{SMV}) \) and \( \text{distance} < \text{min} \_ \text{distance} \) then
12: \( \text{min} \_ \text{distance} \leftarrow \text{distance} \)
13: \( \text{min} \_ k \leftarrow k \)
14: end if
15: end while
16: \( \text{Add}((\text{min} \_ k), K) \)
17: end for
18: return \( K \)
The set of points generated in step (i) contains points belonging to the CCS, but it also contains other points projected from the SMV that are not part of the CCS. In this second step we propose a method to distinguish the points that belong to the CCS from the ones that do not. To this end we use a Random Sample Consensus (RANSAC) [9] algorithm to estimate the parameters of the ideal plane described by the cutting saw in the CCS. This algorithm generates, iteratively, plane models from three random points in the set, and then tries to fit the rest of the points to the model generated. The importance of this fitting stage is that the points are divided into two sets: the inliers and the outliers. If we adjust wisely the threshold that separate these two sets, the proposed method is robust enough to filter the outlier points, returning the largest inlier set of points found. In our case, we have been very stringent with this adjustment since we defined as inlier a point that is closer than 0.1mm from the model plane. The details are listed in algorithm 2.

Algorithm 2 Inlier Selection

1. **Input:** \( K \)
2. **Output:** \( \text{largest\_inlier\_set} \)
3. \( \text{largest\_inlier\_set} \leftarrow \text{empty} \)
4. \( \text{max\_iterations} \leftarrow 10000 \)
5. \( \text{for } i = 1 \rightarrow \text{max\_iterations} \text{ do} \)
6. \( \text{temp\_set} \leftarrow \text{empty} \)
7. \( \text{for each } p \text{ in } K \text{ do} \)
8. \( \text{distance} \leftarrow \text{DistancePointPlane}(p, n, p0) \)
9. \( \text{if } \text{distance} < 0.1 \text{ then} \)
10. \( \text{Add}(p, \text{temp\_set}) \)
11. \( \text{end if} \)
12. \( \text{end for} \)
13. \( \text{if } \text{length}(\text{temp\_set}) > \text{length}(\text{largest\_inlier\_set}) \text{ then} \)
14. \( \text{largest\_inlier\_set} \leftarrow \text{temp\_set} \)
15. \( \text{end if} \)
16. \( i \leftarrow i + 1 \)
17. \( \text{end for} \)
18. \( \text{return } \text{largest\_inlier\_set} \)

Each point included in the inlier set is then associated with the triangular face in the SMV that contains it. This triangular face is then labelled as belonging to the CCS.

At a post-processing stage, a connectivity filter is applied and all the small (one face) holes in the CCS are closed. The result of the algorithm is a set of labelled triangular faces that conform the CCS.

2.4. Validation protocol: Simulated clinical cases

A ground truth for this kind of algorithm would be a SMV with a labelling of the geometry touched by the cutting saw. Since we do not have such a ground truth to test the accuracy of the algorithm, we simulate the cutting process and generate SMVs directly from PMVs. In the simulation we take a PCP, push it along its normal and then vary this normal to a certain degree. This process simulates the offset and angle error that could occur in the operating room.
Figure 2: A simulated resection, showing in semitransparent gray the planned cutting plane and in green the simulated cutting plane (distance offset of 2mm and angle of 30°).

Table 1: Algorithm performance the two cases. The distance and angle parameters are controlled in the simulation. The matched column shows the ratio between the ground truth area and the detected area that matches with it. The not detected column shows the proportion of the area that is in the ground truth but that it was not detected by the algorithm. The not ground truth column shows the proportion of the area that was detected as being part of the cutting surface but it was not part of the ground truth.

<table>
<thead>
<tr>
<th>case</th>
<th>distance</th>
<th>angle</th>
<th>matched</th>
<th>not detected</th>
<th>not ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0.976</td>
<td>0.023</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0.979</td>
<td>0.020</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0</td>
<td>0.984</td>
<td>0.015</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>15</td>
<td>0.981</td>
<td>0.018</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0.978</td>
<td>0.021</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0</td>
<td>0.981</td>
<td>0.018</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0.982</td>
<td>0.017</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0.981</td>
<td>0.019</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0</td>
<td>0.981</td>
<td>0.016</td>
<td>0.003</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0.979</td>
<td>0.006</td>
<td>0.015</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0.992</td>
<td>0.006</td>
<td>0.002</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0</td>
<td>0.986</td>
<td>0.013</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>15</td>
<td>0.994</td>
<td>0.005</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0.988</td>
<td>0.011</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0</td>
<td>0.993</td>
<td>0.007</td>
<td>0.000</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0.994</td>
<td>0.006</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0</td>
<td>0.993</td>
<td>0.006</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>0</td>
<td>0.990</td>
<td>0.008</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Then the PMV is cut generating a SMV and the triangular faces involved in the cut are labelled as belonging to the ground truth. Then our algorithm is run with the SMV and the PCP as inputs. The resulting CCS is matched to the ground truth following these rules:

(i) If a triangular face belongs to the ground truth and to the CCS then the area of that
triangular face adds for the matched ratio (see table 1).
(ii) If a triangular face belongs to the ground truth but it was not detected, the area of that face is added to the not detected ratio.
(iii) The last case occurs when a triangular face is detected as belonging to the cutting surface but it does not belong to the ground truth; in that case, it is added to the not ground truth ratio.

3. Results
The results of this work are shown in table 1. This table shows a comparison of the area detected by the algorithm and the ground truth generated by the simulation. We show that varying the distance offset and the angle of the simulated cutting plane in 0mm, 2mm, 5mm and 0°, 15°, 30° (for each distance offset all the angle variations are generated) does not affect the detection accuracy. The mean matched area detected is around a 98.5% of the ground truth area. The non significance (see figure 4) of the distance offset and angle variations was determined by multi way ANOVA testing, being the p-values for distance offsets of 0.29 and for angles of 0.69 (both >> 0.05). Some qualitative results can be visually assessed in figure 3.
4. Discussion
This work demonstrates the accuracy of an algorithm for cutting surface detection and labeling. The algorithm is useful for surgical intraoperative navigation evaluation, for surgery team performance evaluation and for surgical training. As future work we will simulate the cutting saw vibration noise and we will include real (non simulated) resection cutting path detections, visually validated by orthopaedic oncologic surgeons.

References