Design and Evaluation of Processes for Fuel Fabrication: Quarterly Progress Report #8

Georg F. Mauer
University of Nevada, Las Vegas, georg.mauer@unlv.edu

Follow this and additional works at: https://digitalscholarship.unlv.edu/hrc_trp_fuels
Part of the Nuclear Engineering Commons, and the Robotics Commons

Repository Citation
Available at: https://digitalscholarship.unlv.edu/hrc_trp_fuels/28

This Report is brought to you for free and open access by the Transmutation Research Program Projects at Digital Scholarship@UNLV. It has been accepted for inclusion in Fuels Campaign (TRP) by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
Design and Evaluation of Processes for Fuel Fabrication

QUARTERLY PROGRESS REPORT #8

UNLV AAA University Participation Program

Prepared by: Georg F. Mauer
Department of Mechanical Engineering
UNLV, Las Vegas, NV 89154-4027
Phone: (702) 895-3830
FAX: (702) 895-3936

Reporting Period: June 1, 2003 through August 31, 2003
Design and Evaluation of Processes for Fuel Fabrication

Summary

The eighth quarter of the project covered the following:

- Mr. Richard Silva continued the development of a simulation model with a Waelischmiller hot cell robot. Rich will continue to develop detailed 3-D process simulation models as his M.Sc. thesis project. Rich is employed with Bechtel at the Yucca Mountain project.
- A paper submitted for ANS for the Winter Annual Meeting on hot cell robotics was accepted for presentation and publication.
- Further advances on Concepts and Methods for Vision-Based Hot Cell Supervision and control, focusing on rule-based object recognition (Ph.D. Student Jae-Kyu Lee)
- Undergraduate student Jamil Renno created simulations of fuel pin assembly (robotic insertion of pellets into cladding tubes) for the hot cell manipulator. Besides the correct insertion process, several accident scenarios were simulated.

Part I Hot Cell Manipulator Simulation

During the present reporting period, the robot simulation model for robot control under Matlab Control software was improved further by student Jamil Renno. Matlab controls the spatial robot model, comprising a geometric model as well as the robot dynamics. Thus a realistic simulation of the forces and torques present during robot motion is being generated. Jamil developed a simulation for pellet placement from a bin, and for inserting a row of pellets into a fuel rod. Fig. 1 illustrates the placement of a pellet for insertion. Several accident scenarios associated with pellet placement and insertion were explored and analyzed, see Figures 2 through 6.
Figure 2 Robot Simulation: Accidental dropping of Pellet from Feeder.

Figure 3 Pellets Buckling due to Excessive Pushing Force from the Robot – Start of Buckling
Figure 4  Pellets Buckling due to Excessive Pushing Force from the Robot – Advanced Disarray

Figure 5  Forces during Pellet Insertion.
Simulation Work Plan for September through November 2003

Expand simulations to other processes, such as pellet press: pellet unloading, loading/unloading of pellet trays from sintering oven, grinding, dimensional inspection.
Part II Object Recognition

This section is based on a progress report prepared by Mr. Jae-Kyu Lee. Mr. Jae-Kyu Lee continued his work on recognition using a knowledge-based system. Figure 2.1 shows the image database.

1. Algorithm for Recognition

The schematic diagram for algorithm is shown below:

<table>
<thead>
<tr>
<th>Preprocessing stage (off-line):</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>: Extract interesting points from model image</td>
</tr>
<tr>
<td><strong>Step 2</strong>: Calculate edges and loops data</td>
</tr>
<tr>
<td><strong>Step 3</strong>: Evaluate surface invariance using transformation invariance of mid-point of loops</td>
</tr>
<tr>
<td><strong>Step 4</strong>: In case of 3D, evaluate loop connectivity of super perimeter</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recognition stage (on-line):</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong>: Extract interesting points and build test vector sets</td>
</tr>
<tr>
<td><strong>Step 2</strong>: Pick one model image dataset and compare with test scene data to find co-edges and loops</td>
</tr>
</tbody>
</table>
Step 3: Evaluate DOU values and verify hypothesis

Step 4: Regenerate matching results

Step 5: Go to the step two and pick next model image dataset and repeat the remain steps till the last model

The following describes a series of experiments

Experiment 1: Verify Probabilistic Viewing Effect in 3D

Before we start 3D object, first, we verify the probabilistic viewing effect. To do so we try to find match between the same objects only their image is taken in different viewing angle at 0°, 15°, 30°, respectively. Figure 2.3(a) is recognition between the prism image and itself. The Figure 3(b) is recognition between the Figure 2.3(a) and (b). Figure 2.3(c) is between (a) and (c). The second rows show their junction match after co-edge search from their test vectors. The third rows are final results.
For the first match, recognition between the same images shows extra loops inside. Such unwanted loop match is common in 3D real object recognition due to problems in real image as we previously discussed. This is due not only to round off and environment error but also the errors from viewing itself. Each matching algorithm in 3D has been applied to each surface as well as to the super perimeter. After finding the right match of loops we evaluate the degree of uncertainty (DOU). By following the DOU values, we make a decision which strategy is the best for the processing the recognition. During the evaluation of DOU value we also employ the rank system (Shimshoni & Ponce, 2000). The rank system is also known as weighted $k$th nearest neighbor or WkNN (Duda & Hart, 1973). In our research such weighted value (or so called extra credit) is given when the loop belong to surface also belongs to the super perimeter as well. Such case of matching results will be stored into different dataset by best bin first (or BBF) algorithm (Beis & Lowe, 1998). Table 1 shows the details of the pattern matching process of Figure 2.3.

Table 1(a) The recognition results table shows statistics from various steps of matching algorithm.

<table>
<thead>
<tr>
<th>Figure</th>
<th>Matched junction from co-edge search</th>
<th>Matched loops from junction</th>
<th>DOU value</th>
<th>Scores and decisions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3(a)</td>
<td>11</td>
<td>4 out of 6 from super perimeter</td>
<td>-1</td>
<td>6; perfect match and regenerate super perimeter and its surfaces. No need further search for surfaces</td>
</tr>
</tbody>
</table>

Table 1(b) The recognition results table shows statistics from various steps of matching algorithm.

<table>
<thead>
<tr>
<th>Figure</th>
<th>Matched junction from co-edge search</th>
<th>Matched loops from junction</th>
<th>DOU value</th>
<th>Scores</th>
<th>Decision</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3(b)</td>
<td>7</td>
<td>1 out of 3 from super perimeter</td>
<td>-4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 out of 3 from surface 1</td>
<td>-2</td>
<td>2.5</td>
<td>One of two loop belongs not only surface 1 but also super perimeter we gave extra credit of 50 percent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 from surface 2 and is proven unwanted</td>
<td>-3</td>
<td>0</td>
<td>Number of $</td>
</tr>
</tbody>
</table>
Table 1(c) The recognition results table shows statistics from various steps of matching algorithm.

<table>
<thead>
<tr>
<th>Figure</th>
<th>Matched junction from co-edge search</th>
<th>Matched loops from junction</th>
<th>DOU value</th>
<th>Scores and decisions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 from super perimeter and is proven unwanted</td>
<td>1</td>
<td>-5</td>
<td>Number of</td>
</tr>
<tr>
<td>2.3(c)</td>
<td>2 from surface 1</td>
<td>2.5 one of two loop belongs not only surface 1 but also super perimeter we gave extra credit of 50 percent</td>
<td>-2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.3(c) 5</td>
<td>0</td>
<td>-3</td>
<td>Number of</td>
</tr>
</tbody>
</table>

Experiment 2: Find Model Object from Test Scene C

Here 3D object recognition from a real image is tested. As always, there is no prior knowledge from the test scene except points set coordinates. The procedure is the same as 2D until the loop search. However, after evaluating DOU values, we implement both a super perimeter search as well as a surface search. We compare the highest scores from each step.

1. Generating Test Vector

Figure 2.4 shows a test scene C containing an occlusion. The test vector drawing is the same for a 2D synthetic object recognition.

(a) Test vector drawing from point O  
(b) Test scene C

Figure 2.4. One of eleven test vector sets from test scene A at point O
(2) Find Edge Match and collect Co-edges from Test Vector Sets

Figure 2.5 shows the result of a co-edge search. In Figure 2.5(a), they are many complicated co-edges after match with prism and test scene C. This is due to overlapping, which creates new corner points and removes the original ones. Such ambiguity causes many false positive results even if all objects in the test scene belong to the model image database.

In Figure 2.5(b), the co-edge search result shows better performance. False positive matches have been eliminated because the cubic object in the test scene has not lost its interesting points. However, even if the cubic object is located in front of the prism, the corner detection algorithm usually loses the point P on the surface of prism.

(3) Search for the Loop and evaluate DOU Value

Figure 2.6(a) shows only one loop match for the prism. Figure 2.6(b) shows three loop matches with the cubic object. Therefore the DOU value even for the cubic search from the test scene C concludes that overlapping exists. We use probabilistic reasoning in both cases.

(4) Parallel Process to search for Super Perimeter and Surface

In Figure 2.6, both cases are considered as overlapping. However, since the cubic object is associated with more freshly found loops and those loops are all belong to super perimeter, we directly jump into super perimeter recover, and then collect the surfaces which are belong to that super perimeter. For the prism, the only matching loop also belongs to the super perimeter. Figure 2.7 shows the final result.
Figure 2.7. Final match result
References