Towards Identification of TRISO Pebbles at Arbitrary Orientation Using X-ray Radiographs

M. Stringer, B. M. van der Ende, C. Anghel

Canadian Nuclear Laboratories, 286 Plant Road,
Chalk River, Ontario, Canada, K0J 1J0

Abstract: *Modern pebble-bed reactor concepts employing TRISO-based pebbles involve the circulation of thousands of such pebbles through the reactor core. There is currently no method for the tagging, identification, and tracking of individual TRISO pebbles as they enter and exit the reactor core. Methods for uniquely identifying TRISO pebbles would significantly assist nuclear material accountancy and safeguards as each individual pebble could be tracked within a TRISO pebble reactor. This paper presents progress in demonstrating the use of computer vision software combined with machine learning approaches (Convolutional Neural Networks) for identifying TRISO pebbles from x-ray radiographs, based on the unique distribution of TRISO particles within each pebble. To identify a pebble an x-ray radiograph is compared with a library of reference radiographs of a set of pebbles. The matching performance is affected by the orientation of the radiographed TRISO pebble compared to the reference radiograph. The orientation can be specified by the off-axis and on-axis rotation angles which are defined by the radiography source and detector setup. Translational shifts of the pebble relative to the axis must also be accounted for. Significant progress has been made in the use of computer vision software for identifying any orientation of a TRISO pebble, when including an embedded reference marker, in a given acquired x-ray radiograph. In parallel, complementary machine learning methods are being developed to determine the orientation and to identify the pebble. This hybrid combination of methods can increase the accuracy of the identification. The potential implementation of this method for safeguarding TRISO pebble reactors is also discussed.*

Introduction

Pebble bed nuclear reactors consist of hundreds of thousands of individual pebbles in a single reactor [1]. Being able to uniquely identify individual pebbles is crucial for nuclear safeguards and fuel accountancy purposes. The recognition process must be automated and operate without significant human oversight. One possible option is to take an x-ray image of the pebbles as they exit the reactor. As the pebble consists of a graphite matrix containing the fuel particles, the fuel particles cast shadows in the x-ray image compared to the relatively transparent graphite matrix. The distribution of the fuel particles within the pebble is essentially unique to that pebble as the probability of two pebbles having particles in the same positions is extremely small. Therefore the pebble can be identified by the positions of the shadows in the x-ray image. Previous approaches using machine learning techniques with this method showed good identification performance provided the pebble is orientated in the same way as the reference image [2]. However, relatively small shifts from the reference library images resulted in poor performance. Other techniques have used x-ray computed tomography (XCT) scans to reconstruct the 3D positions of the fuel particles within the pebble [3]. Since the 3D positions of the fuel particle are used, the XCT technique is much less sensitive to the orientation of the pebble compared to the reference image. However, the extended time required to perform an XCT scan may result in operational throughput issues during reactor operation. In this paper we explore the use of computer vision techniques to identify individual TRISO pebbles as they pass through the reactor by comparing x-ray scans of them to a set of reference x-ray images in a library. A Geant4 simulation has been developed to produce a library of reference images and computer vision techniques to identify pebbles based on the simulation results that have been developed. These simulations were used to evaluate the impact of systematic effects, such as rotations of the pebble, on the performance of the identification algorithm. Potential techniques to mitigate the impact of the systematic effects using further computer vision techniques or machine learning techniques have been explored and are also discussed below.

Simulation of TRISO Radiographs using Geant4

The Geant4 simulation toolkit [4][5][6] was used to produce the simulated radiographs of the TRISO pebbles. An image of the simulation of the setup is shown in Figure 1.

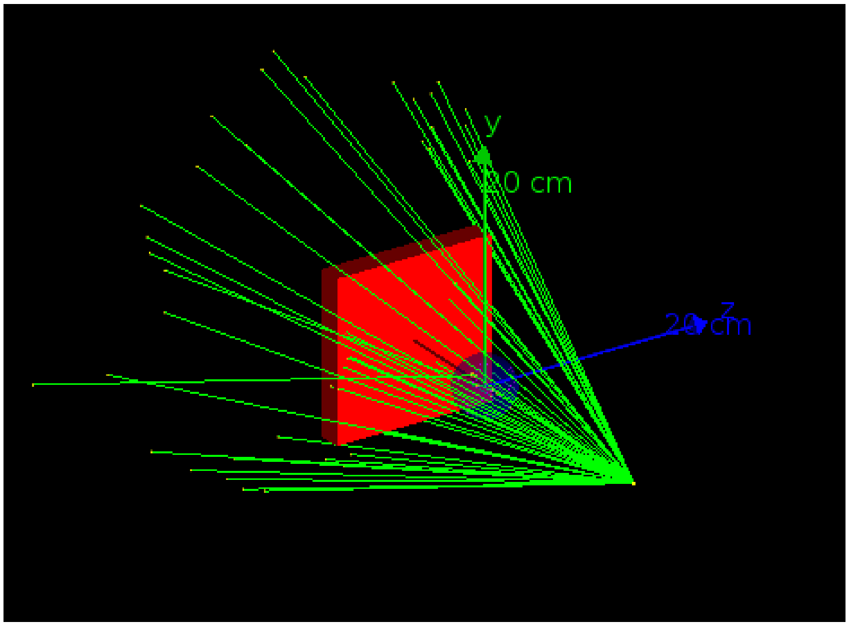

Figure 1: An image of the Geant4 simulation of the x-ray imaging setup. The x-rays (green lines) are generated from a point in a cone towards the red x-ray screen. The blue sphere is the TRISO pebble being imaged. Only a limited number of x-rays have been simulated for illustrative purposes.

Individual fuel pebbles with a layered structure as described in [2] were placed randomly within a graphite sphere. Monoenergetic 150 keV x-rays were generated in a cone towards an x-ray screen[i], any x-rays which reached the screen were binned in a histogram based on their hit position. The binning of the histogram reflected the resolution of the x-ray detector screen.

Several parameters in the simulation are tuneable, including the resolution of the x-ray screen, the objective lens used in the x-ray device, the opening angle of the x-ray beam and the position and orientation of the TRISO pebble. The orientation of the pebble is defined by two angles; the on axis rotation and the off axis rotation. The on-axis rotation refers to rotations of the pebble about the beam axis (the x-axis in Figure 1). The off-axis rotations refer to rotations about an axis perpendicular to the beam direction, (the y-axis in Figure 1). Each individual TRISO pebble is defined by the seed for the random number generator used to position the fuel particles within the pebble.

The parameters of the geometry used in the simulations shown in the following section are listed in Table 1. These parameters were selected to match the setup of an x-ray imaging device at Canadian Nuclear Laboratories as closely as possible. In order to allow for a relatively fast simulation the resolution of the screen was reduced to 512 by 512 pixels which requires less statistics.

Table 1: The simulation parameters used to generate the x-ray images.

Parameter	Value
TRISO sphere radius (mm)	25
Number of particles in pebble	10000
Distance from centre of pebble to objective (mm)	105
X-ray source distance to pebble centre (mm)	85
Beam angle (Degrees)	15
Objective magnification	0.4x
Number of pixels per linear dimension	512
Pixel size (um)	108
Number of x-rays	10^7

A simulation of 10^7 x-rays[ii] incident took 25-35 minutes to simulate on a single core. The histograms were converted to bitmap images representing the x-ray radiographs, one of these images is shown below

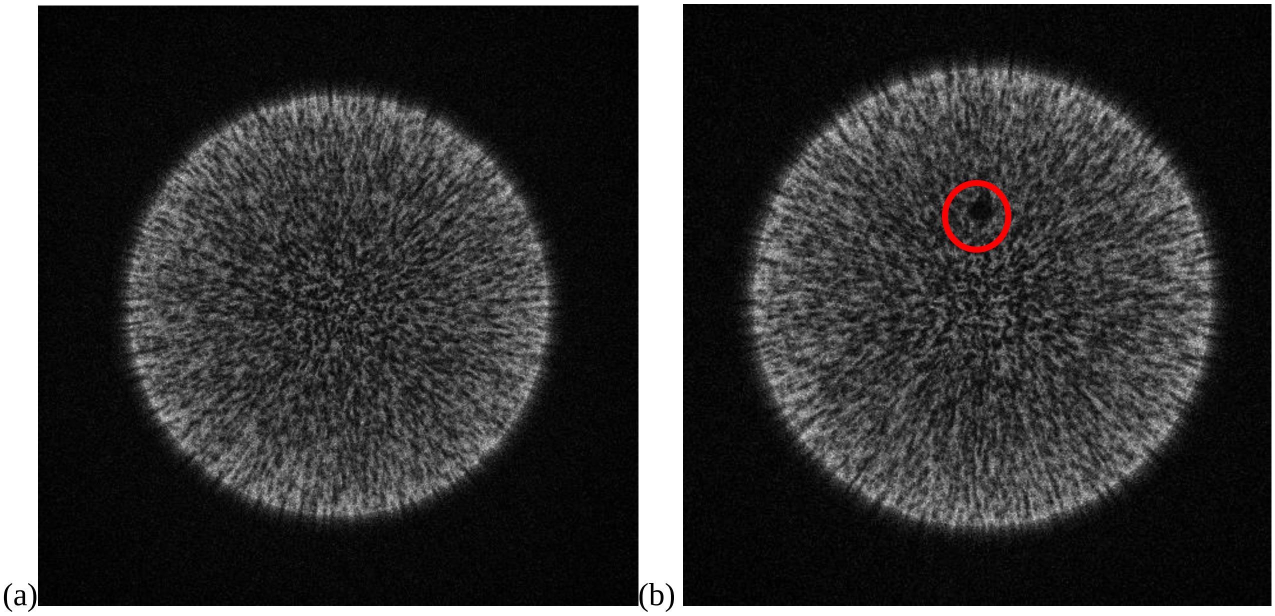

Figure 2: (a) The radiograph of a TRISO pebble produced by the Geant4 simulation using the parameters listed in Table 1. (b) An image of a simulated TRISO pebble with a reference particle (highlighted with a red circle).

Computer Vision Techniques to Identify TRISO Pebbles

The OpenCV library [7] was used to provide the computer vision tools to process these images. To identify a TRISO pebble based on an x-ray imaging, a radiograph image library of all of the pebbles is first built. The library of images is then used to find a match with the TRISO pebble exiting the reactor and being imaged. This algorithm is broadly broken down into three stages: feature extraction, feature matching and determination of the transform on the TRISO pebble.

The feature matching was performed by the ORB (Orientated FAST [Features from Accelerated Segment Test] and Rotated BRIEF [Binary Robust Independent Elementary Features]) feature detector [8]. A feature is a region of an image which is distinct, i.e. one would be able to easily identify the feature in the larger image, and often features are identified at corners of shapes.

Once all the features have been extracted from the library of reference images and the test image; the features from the test image can be compared to the features in each of the reference images. A comparison between each pair of features in the test and a reference image is evaluated. To determine if a match is a good match, the Lowes ratio test is used [9]: a good match is defined as one which matches much better than all the other matches. Matching is often obscured due to random noise. For the following studies, a ratio of 0.75 was used, i.e. the best match Hamming distance (measuring the number of bit substitutions required for a match [10]) must be at least 0.75 of the second best match distance.

The final stage of the calculation is the determination of the transform from one image to another. The transform is defined by the homography matrix [11] which is a three by three matrix defining the mapping of pixels from the original image A=(x,y,1) to a new image B=(x’,y’,1). In the presence of only on axis rotations one would expect the upper left 2x2 submatrix of the homography matrix to look similar to a 2D rotation matrix.

The construction of the homography matrix is done via the RANSAC (Random Sample Consensus) algorithm [12]. The algorithm takes the positions of the well-matched features in both images and fits the parameters of the homography matrix to them. The behaviour of the algorithm is similar to that of a χ^2 fit although it is much more resilient to outliers from incorrectly matched features.

An example of using these functions in the OpenCV library can be found in [13].

The matching of the same pebble at a different orientation is shown in Figure 3. Each individual line represents a pair of well-matched keypoints and the green line is the edge of a square representing the transform from the reference image to the test image. As only an on-axis rotation has been applied to the test image, the homography matrix is essentially just a rotation. No scale or shear is applied to the image.

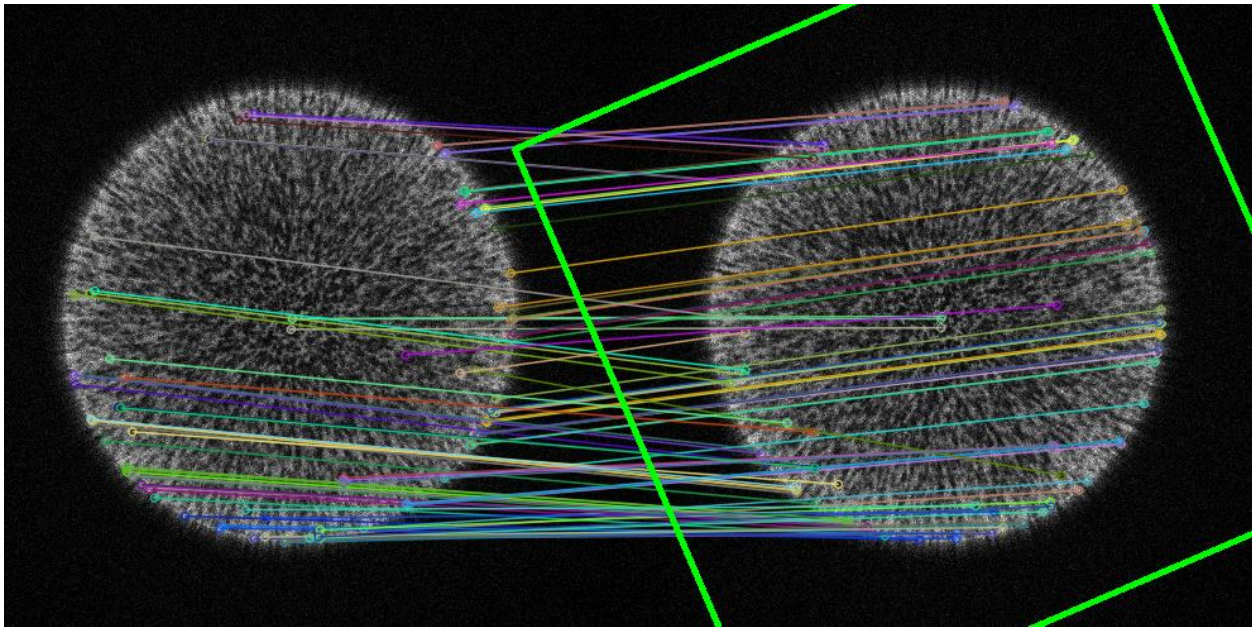

Figure 3: The matching of the same pebble with an on-axis rotation. The colored lines represent good matches between the features in both images. The green box represents the transform from the reference image (left) to the input image (right)

The performance of the matching algorithm when two different pebbles are attempted to be matched is shown in Figure 4. In comparison with Figure 3, there are much fewer matched keypoints, and furthermore they do not match in a coherent way. The homography matrix produced by the positions of the matched features shows significant scaling and shear which is physically impossible.

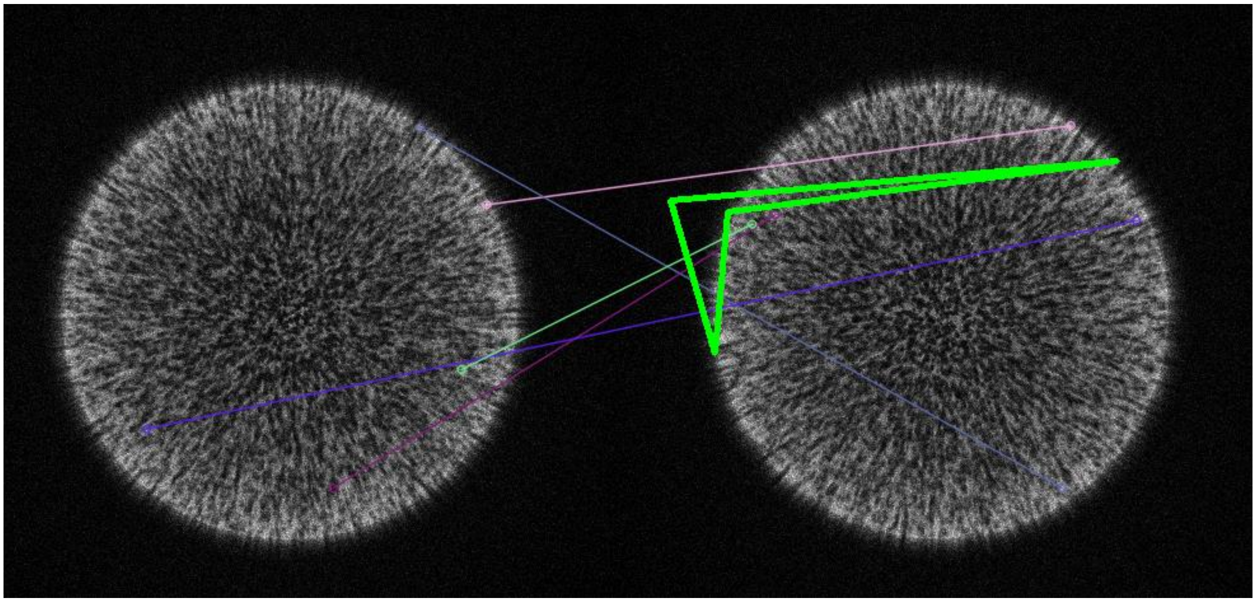

Figure 4: The matching of two different pebbles

Performance of the Algorithm and Systematic Effects

The performance of the algorithm under on-axis rotations is shown in Figure 5(a). The different pebbles dataset and the same pebble dataset each contain 100 images. When comparing the two datasets a total of 10,000 comparisons can be made. When comparing the same pebble dataset to itself a total of 4950 comparisons can be made. For the same pebble the number of well-matched features is on average 100 and ranges from ~50 to almost 200. In contrast when comparing the different pebbles to this dataset less than 10 well matched features are identified. The large separation in the number of matched features indicates that on axis rotations have little effect on the matching capability of the algorithm and it is incredibly unlikely that a pebble will be misidentified as another.

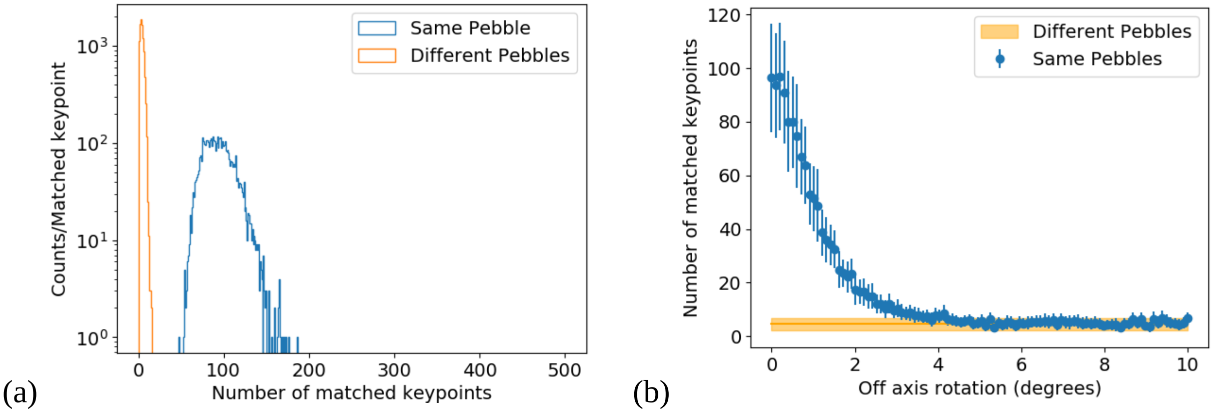

Figure 5: (a) A histogram of the number of matched features for simulations of the same pebble with random on-axis rotations and different pebbles. (b) The number of matched features as a function of off-axis rotation.

Simulations of the same pebble with various off axis rotations were compared to the on-axis rotation dataset. The results of this comparison can be seen in Figure 5(b). The number of keypoints rapidly drops off with off-axis rotation angle: at two degrees only 20 features are matched, and beyond approximately four degrees the matching performance is equal to that of different pebbles In other words, it is impossible to identify pebbles if their orientation is off from the reference image by more than 4 degrees.

Orientating the Pebble by Inserting a Reference Particle

The current image matching code is insensitive to on-axis rotations of the pebble, however off-axis rotations beyond a few degrees pose a significant problem for identifying the pebble. As the spherical pebbles can exit the reactor in any orientation, it is very unlikely that they will be lined up with any of the reference images and any attempts at identifying the pebble will be unsuccessful unless the pebble is orientated before the matching takes place. One possible option to orientate the pebble is the addition of a reference particle to the pebble. The reference particle would be distinct in size from the fuel particles in the pebble and provide information on the orientation of the pebble based on its size and position. The reference particle was chosen to be made of Tungsten, with a diameter of 1 mm, and an offset of 1 cm from the center of the pebble. An example of a simulated radiograph with a reference particle can be seen in Figure 2(b).

Two approaches were taken to determine the orientation of the pebble. The first uses standard computer vision tools to find the position and size of the reference particles shadow in the image whereas the second used machine learning techniques to determine the orientation of the pebble.

Orientating the pebble using computer vision techniques

The computer vision procedure uses two steps to determine the orientation of the pebble. The first is the determination of the position and size of the reference shadow in the image via template matching. The second stage is the transform from the size and position of the best matched template to the off axis rotation of the pebble.

An example of the matching is shown in Figure 6(a). The green circle is the region where the template matching occurs inside, it is determined by the maximum possible offset of the reference particle shadow in the image. The blue circle is the best matching template. It should be noted that the resolution of the image had to be increased to 2048x2048 pixels and the number of x-rays incident on the pebble had to be increased to 10^9. Whilst this significantly increased the runtimes of the simulations, this is still significantly less than the number of x-rays used in a real world scan of a pebble.

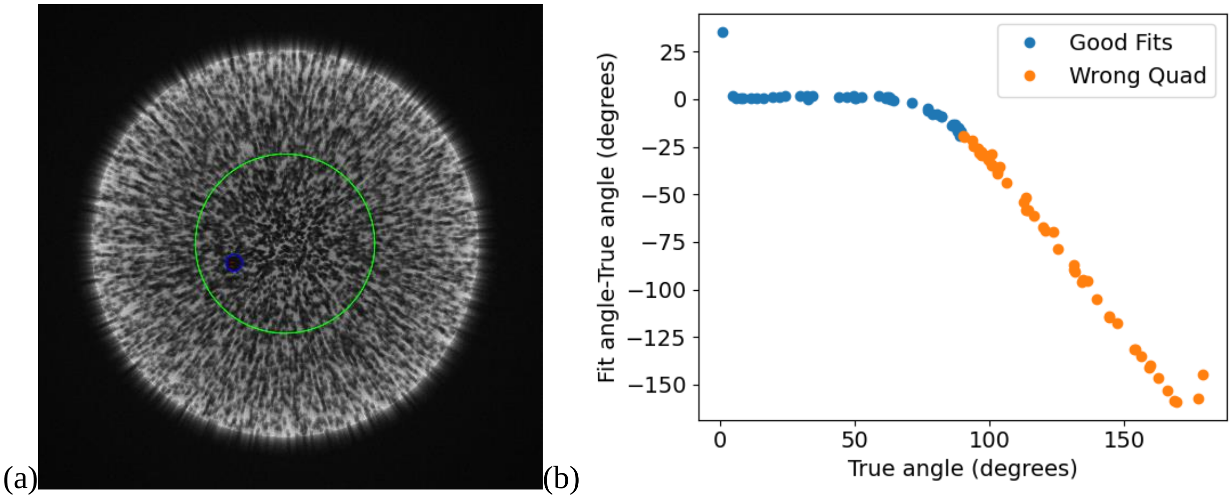

Figure 6: (a) Example of the template matching algorithm used to find the position of reference particle in the image and determine the orientation of the pebble. (b) The true off-axis rotation angle compared to the calculated off-axis rotation using the template matching algorithm. Points labelled Wrong Quad were deduced to be in the incorrect quadrant (i.e. the true off-axis rotation is greater than 90 degrees but the fitted off-axis rotation is less than 90 degrees)

The off axis rotation can then be determined geometrically using the position and the size of the shadow cast by the reference particle.

The matching performance is shown in Figure 6(b). The code produces highly accurate fits up to approximately 60 degrees. Beyond this, the code begins to underestimate. The code has incredibly poor performance at determining identifying angles beyond 90 degrees and is strongly biased towards smaller off-axis rotations. The reason for this is thought to be due to the overlap between the shadows due to the fuel particles and the reference particle as the overlap may make the shadow appear larger than it actually is and therefore the larger templates (corresponding to the smaller off axis angles) fit better resulting in a significant under estimation of the off axis rotation. Further work is underway supplementing these x-ray images with others in order to more accurately determine the orientation of the pebble.

Orientating the pebble using machine learning techniques

The same Geant4 simulations used to test the computer vision algorithms were also used to train a neural network to orientate the pebble based on the x-ray images of the pebbles with a reference particle inserted. The dataset consisted of 1200 images of 120 unique pebbles where 10 different orientations of each pebble were simulated. The entire dataset was divided into training, validation and test datasets. The training set consisted of the images of 80 of the pebbles. Both the validation and test datasets contained the images of 20 pebbles each. A supervised learning technique was used, where the input is the simulated x-ray image and the target is the off-axis or on-axis rotations known from the simulation inputs. The neural network architecture in both cases was a standard ResNet18 network, followed by two fully connected layers to predict either the off-axis or on-axis angle, a single scalar output.

The performance of the neural network after 200 epochs on the test dataset is shown in Figure 7. The network performs well at identifying the on axis rotations of the pebble. Most points lie near to the *y* = *x* dashed black line, meaning that the predictions are nearly equal to the actual values for most images. The images where the errors are larger correspond to those where the off axis rotation is near to 0 or 180°, when the reference particle is near to the centre of the image, no matter what the on axis rotation is. The performance for the off axis rotations is less accurate especially for angles close to 90°. For these, the predictions lie close to the red dotted line *y* = 180° - *x*. The reference particle appears in nearly the same position on the x-ray image for off-axis angles of *x* and 180°- *x*, although the intensity may be different. Further work is in progress, training the neural network on higher resolution images with a higher number of simulated x-rays to see if the performance can be improved.

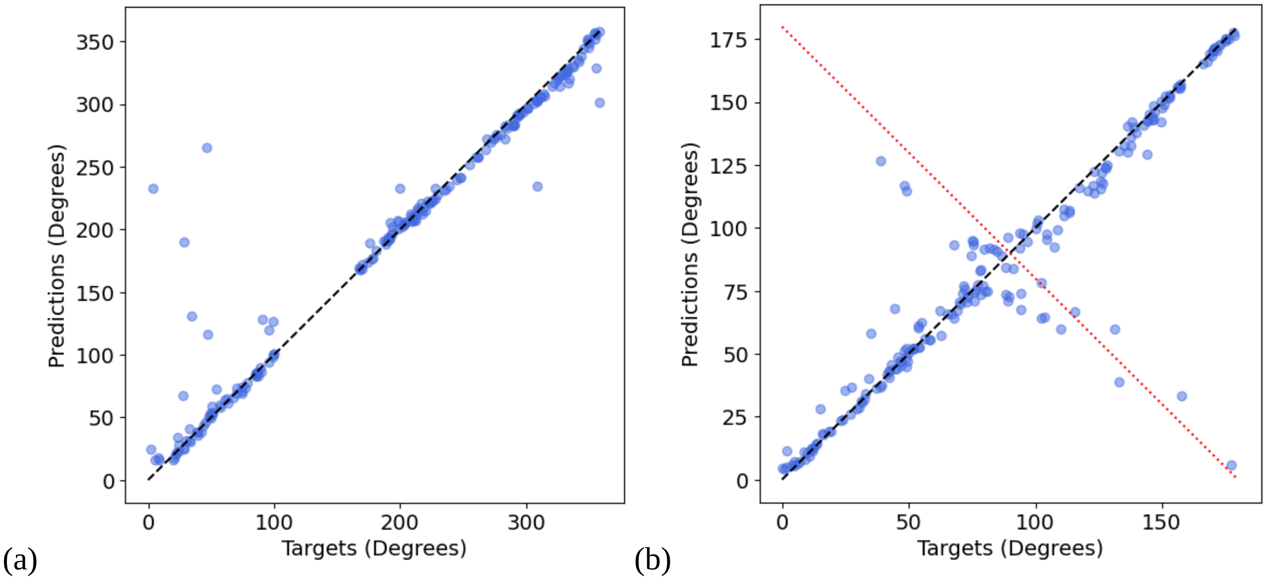

Figure 7: The performance of the trained neural net on identifying the (a) on-axis and (b) off-axis rotations for the images in the validation dataset.

Conclusions

The performance of a standard computer vision algorithm has been investigated when matching x-ray images of TRISO pebbles to a library of pebbles. The algorithm showed good matching performance provided that the "off-axis" rotation is close to that of the reference image. The method is unaffected by "on-axis" rotations of the pebble. The algorithm was first tested on simulated radiographs produced using the Geant4 toolkit. These algorithms have the advantage over machine learning techniques described in [2] as they do not require extensive training.

Work is underway to orientate the pebble by inserting a Tungsten reference particle into the pebble. The addition of the reference particle introduces an additional shadow into the x-ray radiograph. The orientation of the pebble can then be determined by the position and size of the shadow in the image. Two approaches are being used. The first uses standard computer vision techniques, whereas the second uses a machine learning approach. Additional work investigating the addition of a second scan of the image zoomed in on the reference particle has also commenced. It is hoped that the additional image will provide further information such that the pebble can be re-orientated more accurately and the matching algorithms can be applied successfully.

Acknowledgements:

This research made use of the resources of the High Performance Computing Center at Canadian Nuclear Laboratories. This work was supported by the Federal Nuclear Science and Technology program, under the auspices of Atomic Energy of Canada Ltd.

References

[1] A.C. Kadak A future for nuclear energy: pebble bed reactors, *International Journal of Critical Infrastructures 2005 1:4, 330-345*

[2] E. H. Kwapis, H. Liu, K. C. Hartig, Tracking of individual TRISO-fueled pebbles through the application of x-ray imaging with deep metric learning, *Progress in Nuclear Energy, Volume 140, 2021, 103913, ISSN 0149-1970*

[3] M. Fang, A. Di Fulvio, Algorithms for TRISO Fuel Identification Based on x-ray CT Validated on Tungsten-Carbide Compacts, *Apr 28, 2022, e-Print: 2204.13774 [physics.ins-det]*

[4] J. Allison et al., Recent Developments in Geant4, *Nucl. Instrum. Meth. A 835 (2016) 186-225*

[5] J. Allison et al., Geant4 Developments and Applications, *IEEE Trans. Nucl. Sci. 53 (2006) 270-278*

[6] S. Agostinelli et al., Geant4 - A Simulation Toolkit, *Nucl. Instrum. Meth. A 506 (2003) 250-303*

[7] Bradski, G. (2000) The OpenCV Library. *Dr. Dobb's Journal of Software Tools, 120; 122-125.*

[8] E. Rublee, V. Rabaud, K. Konolige, G. R. Bradski: ORB: An efficient alternative to SIFT or SURF. *ICCV 2011: 2564-257*

[9] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints, January 5, 2004

[10] OpenCV Feature matching with flann tutorial, https://docs.opencv.org/3.4/d5/d6f/tutorial_feature_flann_matcher.html (Accessed 21/03/2023)

[11] G. Roth, Homography Lecture Noteshttp://people.scs.carleton.ca/~c_shu/Courses/comp4900d/notes/homography.pdf (Accessed 21/03/2023)

[12] K. G. Derpanis, Overview of the RANSAC Algorithm, http://www.cse.yorku.ca/~kosta/CompVis_Notes/ransac.pdf (Accessed 21/03/2023)

[13] OpenCV object matching Tutorial, https://docs.opencv.org/3.4/d7/dff/tutorial_feature_homography.html (Accessed 21/03/2023)

[14] OpenCV template matching Tutorial, https://docs.opencv.org/3.4/de/da9/tutorial_template_matching.html (Accessed 21/03/2023)

[i] In the simulation the x-ray screen actually serves as the image of the x-ray screen as viewed through an objective lens

[ii] 10^7 x-rays is several orders of magnitude less than the number of x-rays produced in a real world scan