

DIGI-MAT MINI-LAB: FINDING ATOMIC DEFECTS THREE WAYS  
(developed by C.-H. Lee and P.Y. Huang)

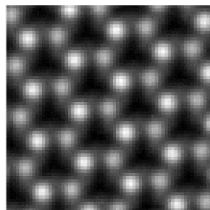
In this exercise, you'll use different methods to identify atomic defects in a 2D material: manual hand labeling, a very basic intensity thresholding, and a convolutional neural network. We will be using atomic resolution images of Te-doped WSe<sub>2</sub> taken by Chia-Hao Lee in the Themis microscope in the Materials Research Laboratory. As we discussed in the lecture, these images contain several different types of point defects including vacancies and substitutions. You can see our full paper on this work here: <https://pubs.acs.org/doi/abs/10.1021/acs.nanolett.0c00269>

### Section A: Manual Labeling

To appreciate the advantages of advanced data processing methods, I think it is important to see how things were done before the advent of fast, readily accessible neural nets. Even today, the gold standard in my opinion for identifying defects is human labeling. Let's measure the speed and accuracy of your brain for identifying atomic defects.

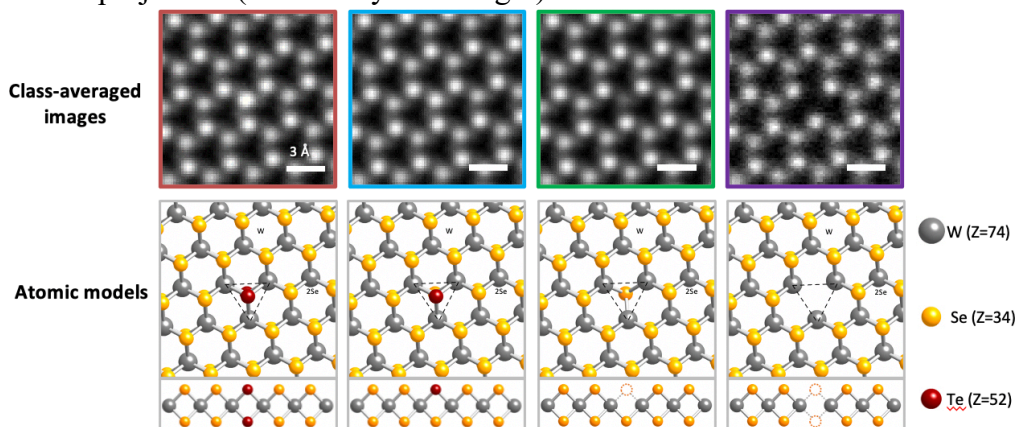
#### 1. Record the time right now

- Below is a set of images to train your brain on what the four most common defects look like in this sample. First, here is what the crystal looks like if it does NOT have a defect:



In this image, the brightest dots are individual tungsten atoms, and the slightly dimmer dots are two selenium atoms on top of each other.

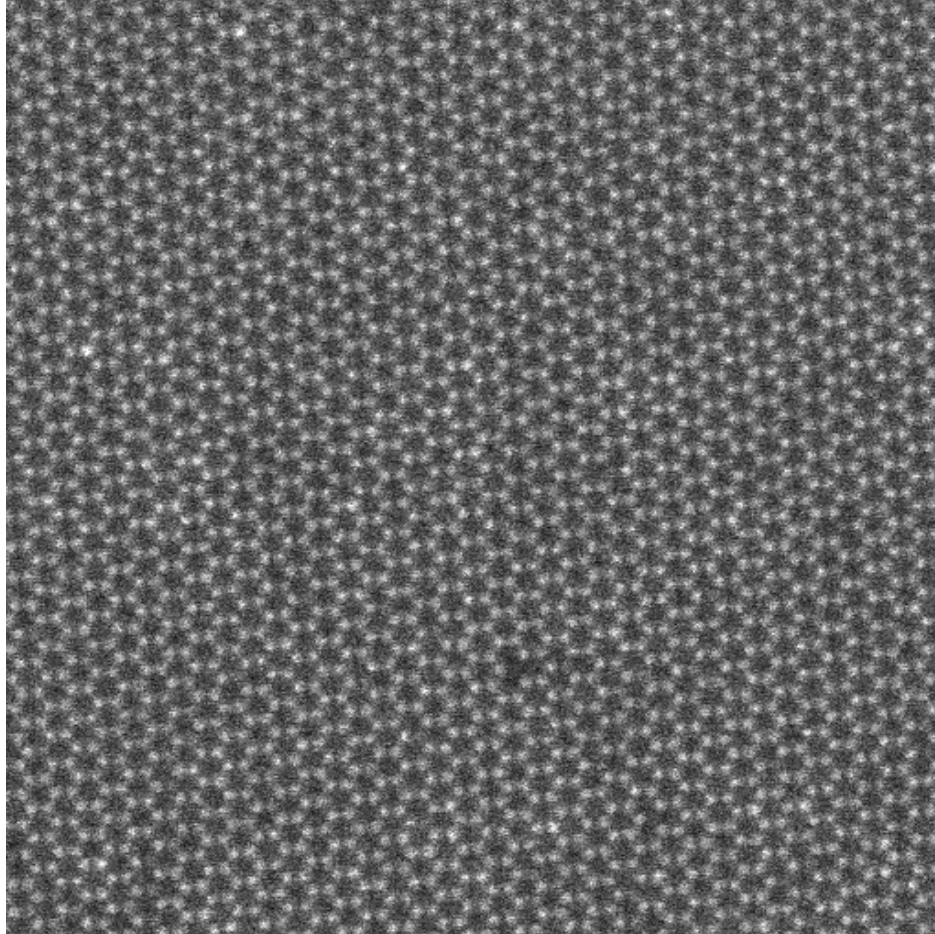
- Now, here are the four most common defects we saw, along with models of their structures. In each image, the defect is the center atom. Focus on the two defects on the right: the "SV" single vacancy, where the atomic column appears a bit dimmer than normal (second from the right) and the divacancy where there is no atom in projection (all the way to the right).



Now, it's time for you to test the mental model you have developed.

**2. Record the time right now**

3. Find all of the single and double vacancies in the image below and circle or mark them in some way. I threw the image in Powerpoint and drew circles around the defects there, or you can print it out. You don't have to distinguish the two defects, just mark both. Your operation should "time out" after 5 minutes (i.e. **don't spend more than five minutes on this!**)

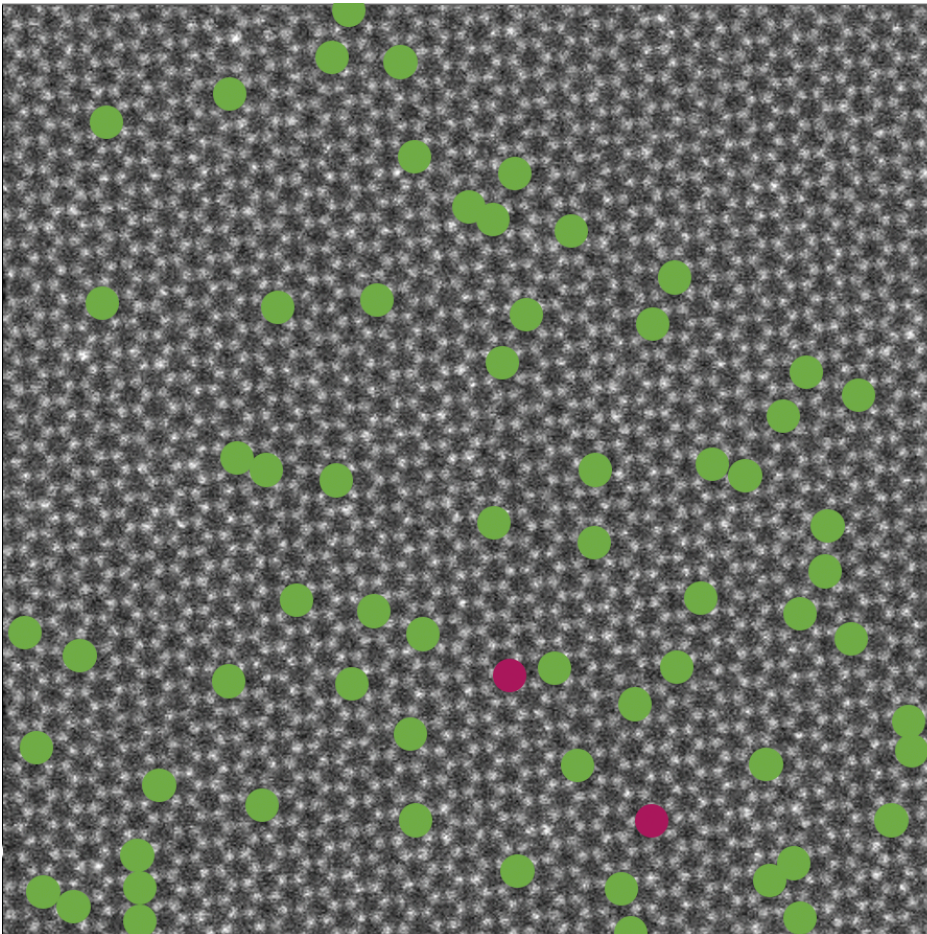


**4. Record the time right now.**

5. Calculate how long it took for you, in seconds, to complete this task.

6. Next, we will calculate the accuracy of your brain's neural network in finding point defects in  $\text{WSe}_2$ . Below is the same image, now overlaid with my labels. Green dots represent single vacancies and magenta dots are divacancies. We will assume this is the ground truth. Compare it with your results from #2. Count the number of True Positives, False Positives, and False Negatives for your identification of Se vacancies. The true positive is defined as the number of defects you identified which were real (see table for more information). For example, if you found 10 Se vacancies and 7 were accurate, 3 were not vacancies, and 2 more vacancies existed that you did not identify, then  $TP=7$ ,  $FP=3$ ,  $FN=2$ . To help you calculate the number of false negatives, there are 66 vacancy defect positions in the image below.

	Predicted Yes	Predicted No
Actual Yes	True Positive (TP)	False Negative (FN)
Actual No	False Positive (FP)	True Negative (TN)



7. Use your answers from above to calculate the recall and precision of your model. Recall and precision are two common metrics used to quantify the performance of classification algorithms. The closer the value is to 1, the better.

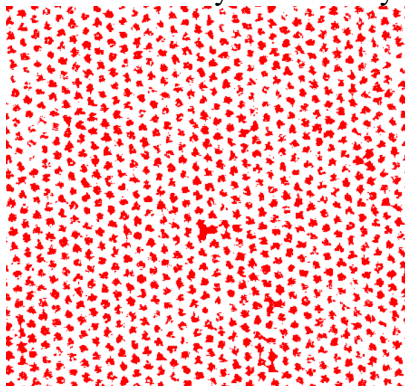
$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

### Section B: Intensity Thresholding

Of course, electron microscopists do not rely solely on processing data by hand. It is common for us to use a variety of sophisticated digital analysis methods to extract useful information from our data. In this section, you'll use one of the most basic tools, intensity thresholding, to pick out vacancy defects.

- First, download ImageJ, a widely used and powerful freeware image processing software. ImageJ and its associated scripts are well documented online. An ImageJ tutorial is available at: <http://imagej.nih.gov/ij/docs/index.html>. There are several versions of ImageJ available; they come bundled with different pre-installed packages. I suggest FIJI, which you can download here: <https://imagej.net/Fiji/Downloads>
- Next, download the image on the course website and open it with ImageJ. Because we're using digital image processing now, we are using an image that is a bit larger. For Section A, you were processing the top right quadrant of this image.
- Smooth the image to reduce the impact of noise. Go to Process → Smooth.
- Threshold the image to select the vacancies in red. Select Image → Adjust Threshold. Move the sliders on the image intensity histogram until most of the defects are covered as completely as possible in red. It's OK if the threshold selects objects other than defects as well, as long as they appear different in size. You can see how I set my threshold below in this small sub-region—see how the divacancies show up as trefoils? With some luck, you can pick up single vacancies this way too. When you're ready, apply the threshold.



### **8. Record the time now**

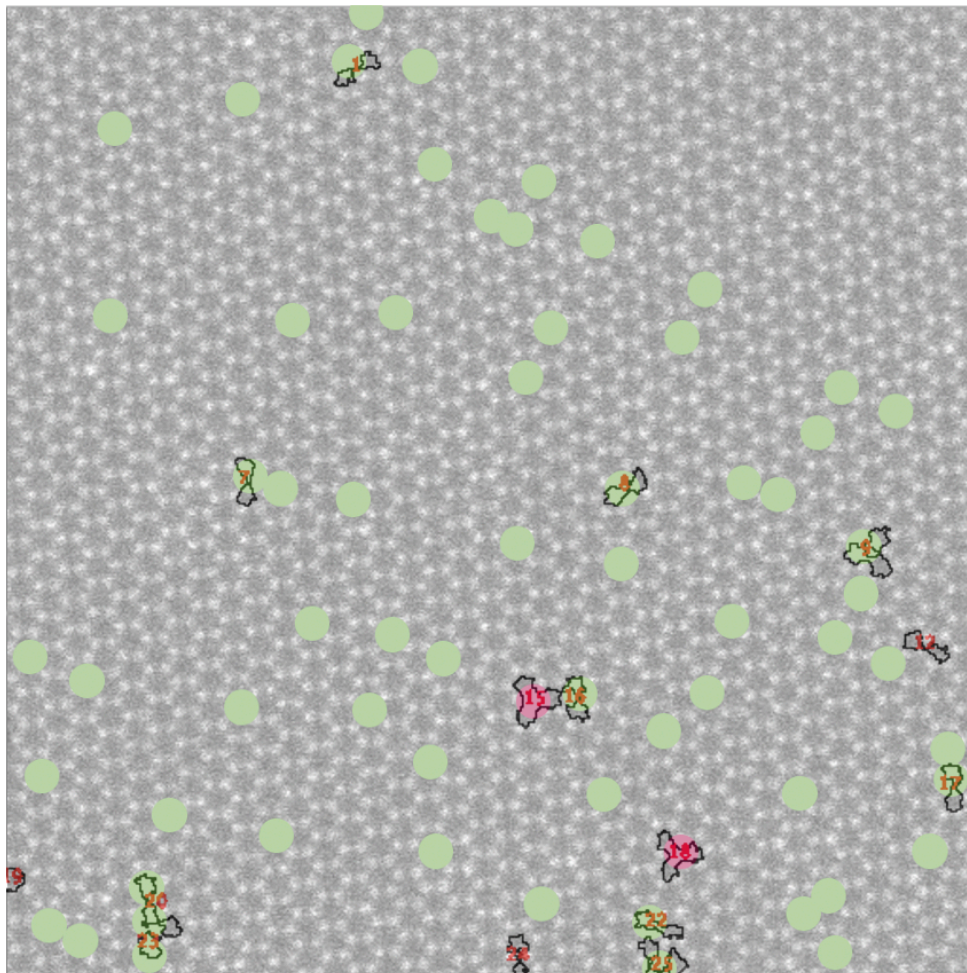
Now go to Analyze → Analyze Particles. In the first box, set the minimum size to be large enough so that only vacancies are selected. You can leave the maximum size at infinity or adjust it if you want. Select Show: Outlines, and Display Results, Clear Results, and Exclude on Edges. Click OK to measure. This should

pick out some of the single and double vacancies. You can repeat this multiple times, inspecting the outlines produced until you are happy with the result.

**9. Record the time now**

10. Report the time it took, in seconds, to locate defects using this method

11. Compare the defect measurements from ImageJ with the ground truth. The top right hand corner of this image corresponds to the image from Section B. If you're having trouble comparing the images, save the output image and overlay it with the ground truth in Powerpoint or something similar. My comparison map is below. Report the Recall and Precision scores for Intensity Filtering for detecting single vacancies.



12. Discuss: Is this method better at locating single or double vacancies? Why?

13. Compare your results to a friend/neighbor; do your results differ significantly?

### Section C: Convolutional Neural Network

Finally, we will use convolutional neural networks to locate defects in  $\text{WSe}_2$ . Luckily for you, we have already trained the neural network to recognize single vacancies in this system! Follow through the notebook at the link below. Work through sections 1-5. You can also take a look at section 6, which creates an averaged image of the defects, if you have extra time

[https://colab.research.google.com/github/chiahao3/EM-scripts/blob/master/DIGIMAT\\_defect\\_finding.ipynb](https://colab.research.google.com/github/chiahao3/EM-scripts/blob/master/DIGIMAT_defect_finding.ipynb)

14. Locate in the notebook the time it takes to process and locate the single vacancies. Note there are two models running in the notebook, which can locate two different defect types. Report the total time cost for the SV model, which locates single vacancies.
15. Now, compare the time required as well as the precision and recall for all three methods you have used (hand labeling, intensity filtering, neural network) and discuss the advantages and disadvantages for each method. For the convolutional neural network, we'll just use the values reported in the paper for the SV model:

<b>Recall: 0.9956</b>	<b>Precision: 0.998</b>
-----------------------	-------------------------

16. Discuss the advantages and disadvantages of each method, using your answers to the previous question. For this question, consider not just the time it takes for each model to run, but also the time required for setup and training. For example, in Section A, you should be able to figure out both the time it took to run your mental model and the time it took you to train it. For the neural network used in Section C, it takes us a few weeks to build a new model, including the time it takes to create labeled training data, train and optimize the neural net, and evaluate the results.