CAPSTONE PROJECT: FIRST PROGRESS REPORT

Section 1. Description of the problem and the data set

1. Background & Problem Statement

Bacteria-related illnesses are responsible for approximately 5 million deaths per year worldwide affecting not only the developing world, but also becoming more common in the developed world with increasing instances of food contamination, hospital-acquired infections, and bioterrorism.

Identification and monitoring of bacteria outbreaks and antimicrobial resistance is critical for tracking health developments and is recommended by the WHO. Current microbial identification approaches can be extraordinarily time consuming, laborious, and expensive. Specifically, in the case of gold-standard clinical microbiology assays, such as staining and microscopy following culture on solid media in Petri-dishes (shallow cylindrical glass or plastic lidded dish that biologists use to culture cells), the assistance of experts for morphological identification is required. Furthermore, these identification methods are often sample destructive.

Thus, *the development of a rapid, automated process to identify and characterize bacterial species from environmental and clinical samples would be a major health innovation*.

2. Project Goal

The aim of the project is to develop a machine-learning based approach for the identification and characterization of bacteria via analysis of photographs of bacteria colonies from environmental samples grown on solid media in Petri dishes, i.e., building a model that could classify colonies on a plate into species, based on aspects such as colony morphology, color, and other features. In a longer term, we may also have direction of simulating the growth of bacteria, or to identify the neighborhood environment based on the photographs of one specific bacteria.

3. Descriptions of Dataset

The dataset consists of pictures of bacteria colonies collected from the soil on 10 different locations (referred to as locations 1-10) in various parks in Manhattan. Because of different environments and neighborhoods, same colonies in different parks may appear differently. But we believe that we could identify some properties based on its species.

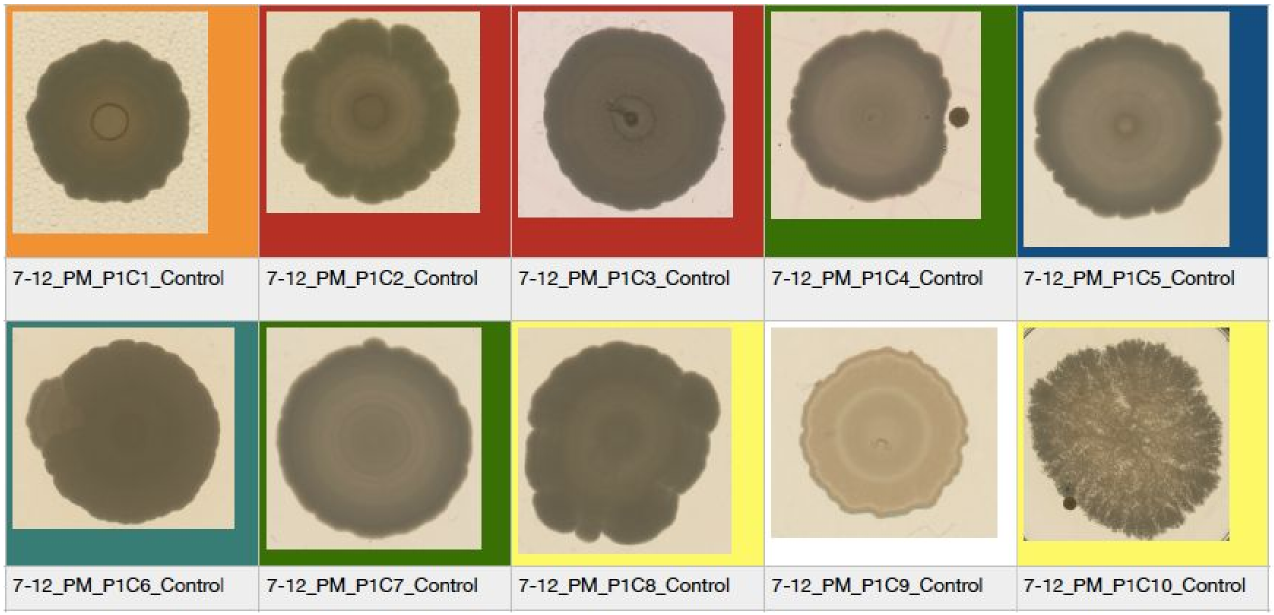
Data was then organized on different folders corresponding to 3 different types of images taken at 8 different time, explained as follows:

- Type “**Serial**” corresponds to images of a single petri dish containing colonies of the same bacteria but at 9 different concentrations, which were allowed to grow over time.

- Type “**Control**” corresponds to images where a single drop of a bacteria species was dropped in the petri dish at a specific concentration (instead of 9 different concentrations like in the case of “**Serial**” type). This was done with the purpose of checking if having several colonies in the same petri dish has any influence in the growing patterns vs having a single colony so the algorithm could learn from both situations
- Type “**Streak**” corresponds to multiple drops but is where the streaks (linear pattern) is drawn across the petri dish with pipettes with different concentrations.

For each of the 3 types described above, images were taken during 4 different days, 2 collections each day (AM and PM, 12 hours apart), for a total of 8 different timestamps. This will be useful for the bacteria classification model to learn different patterns of colonies form the same species, over different growing times.

Colonies for each location were classified using conventional classification methods described above and they were found 7 different bacterial species. Picture 1 shows type “**Control**” images taken at the same timestamp, where each image corresponds to each of the 10 different locations and colors represent different species, so we can see that locations 2 and 3 correspond to the same bacterial species, same that locations 4 and 7 and locations 8 and 10:



Picture 1. control type images of 10 different bacteria colonies, each corresponding to a different location. Colors represent different bacterial species.

For our initial study, we will be focusing on “Serial” and “Control type”.

Section 2: Initial Data Exploration

Data Summary and Data Exploration::

The total of available images: 200 ; Original image size: (4000, 3200, 3)

Location	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Total	20	20	20	20	20	20	20	20	20	20

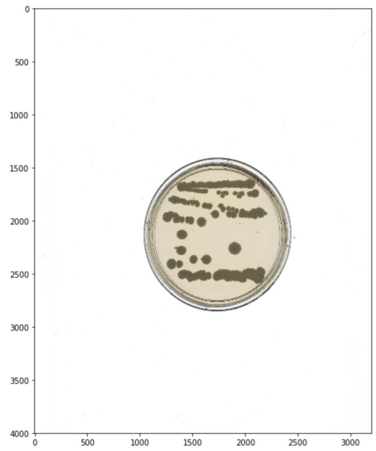
Since the dataset consists of only 200 images, 20 images per location, there is not much opportunity for data exploration, so we are including a description of our Data Preprocessing work instead.

Data Preprocessing:

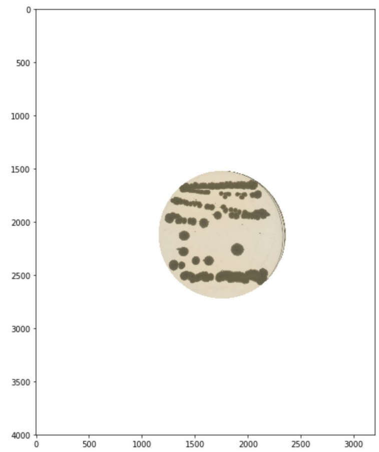
In order to perform data augmentation and at the same time try to create a model that learns to generalize, we cropped the images into small patches so the model can learn to predict classes from patches containing just parts of the original petri dish images. For this purpose, we worked on the following steps:

Step 1: Filter out the petri dish boundaries

For possible image segmentation (identify which pixels correspond to which bacterial species), we need to create masks to identify positive image regions (regions in the image containing pixels corresponding to bacterial colonies instead of just petri dish background). The image backgrounds are generally uniformly distributed which makes creating such masks an easy job. However we still need to filter out the petri dish boundaries because the dark lines of those can be mistaken as positive patches (image segments with effective objects). Considering the shape, size, and center of the petri dish boundaries in the images are similar, we used a circular mask to filter out the boundaries:



Picture 2a. Original Streak Image



Picture 2b. Image processed with mask

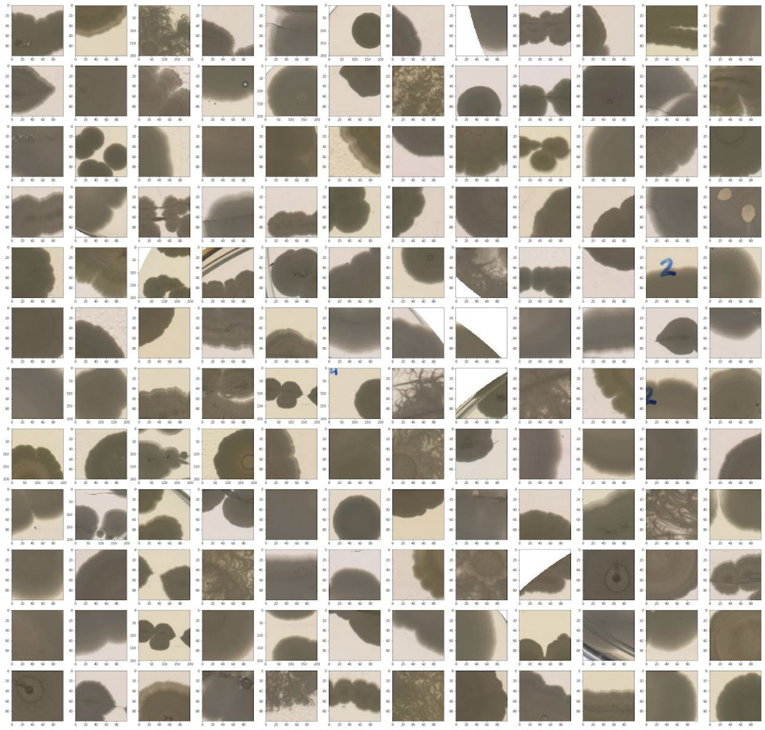
Compared with picture 2a, picture 2b is processed with cleaner mask. However, some pictures are away from the empirical centers, and in that case, part of the images will be compromised. In future, we need to use edge detection to automatically locate the center and radius.

Step 2: Create patches

We used a sliding window with size 100*100 and step size 50 to go through each image and the outputs are the patches. The window size directly determines the size of the patch, which is an important parameter to tune in later work. At present, we only used one window size to create patches which is out of the convenience to test the validity of the preprocessing module. In practice, we will create different size patches which enables model to learn features from a higher level and use validation data to evaluate the performance of the model using different patch sizes.

Step 3: Filter out the negative patches

We can see from the picture 3 that the majority of the images are just blank (background pixels), so the majority of the patches are also blank which are considered negative because no object appears in the patch. To filter out the negative patches, we need to transform the image to grayscale and sum up the pixel of the patch. If the pixel sum is higher than a threshold we set, the patch is considered as positive. In the experiment, we set the threshold as 0.3 (the ratio of objects in the patch to the background). The threshold can be further tuned to best fit the prediction model. We randomly select 150 patches and check their validity. As shown in the picture 3, all patches contain parts of the object, which is a good start.



Picture 3 Randomly selected 150 positive patches

Summary of initial patches with current patch configurations

The total of available positive patches: 8757

Location	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Serial	427	378	239	382	305	452	355	684	0	223
Control	66	66	44	124	112	53	132	91	18	181
Streak	833	499	177	370	240	613	276	1078	0	339
Total	1326	943	460	876	657	1118	763	1853	18	743

As shown in the table above, Location 9 has so few patches in the training dataset. The main reason is because of the circular mask. In general, the center and radius of petri dishes in the images are similar, and that's why we used a fixed center and radius for circular mask. However, location 9 may have a very different petri dish location, so the mask did not cover the object. This problem needs to be fixed in future by automatic center detection.

Important parameters

Main parameters we need to tune in the data preprocessing module:

1. Center and radius of the circular mask
2. Window size and step size
3. Threshold between negative and positive patches

Next step for data preprocessing

Because the mask radius and center are hard coded at present, the boundary filtering process in step 1 can be very fragile once the format of the image chances. In this case, the model would be more robust if it can automatically decide the center and radius of masks for different images.

Section 3: Literature Review

Last decade has seen remarkable developments in the field of building image classifiers using computer vision. The era of deep learning started with the success of AlexNet (Russakovsky et al., 2014)[1] on the benchmark computer vision dataset, ImageNet[2] focused on studying natural world images. This result was steadily followed by development large and deeper neural network architectures in coming years. These architectures saw remarkable improvement in model performance in contrast to any other traditional methods that involved hand crafting features. By 2015, models like ResNet (He et al., 2015)[3] were performing better than a human baseline. This motivated researchers to make similar progress on even difficult tasks such as image segmentation.

Biggest breakthrough of computer vision algorithms for image segmentation came in the form of U-Net[4] which was designed for solving the challenge of cell tracking. The advantage of the method stemmed from its ability to learn from limited training data making it all the more effective for a wide variety of biomedical tasks.

The problem of using automated computer vision algorithms for the purposes of improving diagnosis from medical images is not a new phenomenon. Researchers have long been interested in applying the latest advancements in machine learning to improve the accuracy of their diagnostic procedures and achieve quick and reliable results.

Rajpurkar et al.[6], used a variant of state-of-the-art DenseNet (Huang et al., 2016)[7] algorithm for exceeding human level performance on the task of detecting pneumonia by looking chest X-ray images. By visualizing the internal layers of the network, researchers were able to understand which the portion of images were responsible behind activation of the classifier. Another example for using medical image to improve diagnosis comes studying detecting diabetic retinopathy from retina scans[8].

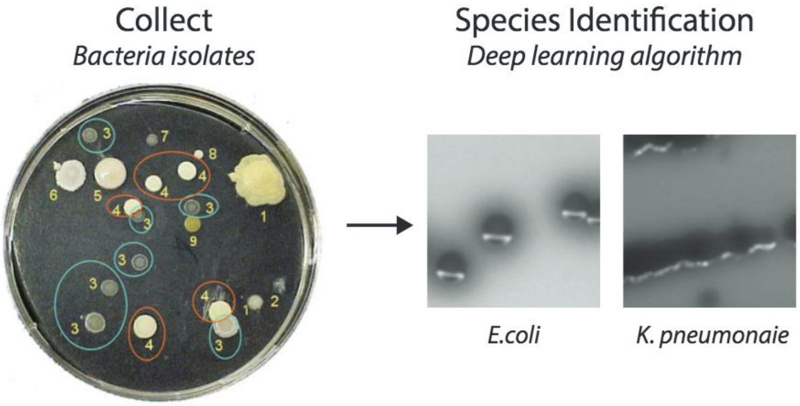
In a collaboration between researchers at Google and a team of dermatologists[9], a model was developed that employs using convolutional neural networks as image feature extractors and combines them with the other available information from patient’s medical history to predict the existence of a certain skin condition. This hybrid approach is designed to utilize the available metadata along with raw images to incorporate more information while building a classifier.

Huang et al.[10] proposed the work to automatically identify bacteria morphology which is the first step towards identifying bacterial species before passing them through additional identification processes. Their research was primarily inspired by the AlexNet architecture applied to the task. They also used unsupervised approaches like autoencoders to reveal novel characteristics from raw images that can help the clinical staff.

Zieliński et al.[11] described using convolutional neural networks as feature extractors and then further building a support vector machine model on top of it for classifying bacterial colonies. The data used in this study comprises 660 high resolution of coloured digital images of 33 different genera of bacteria. The motivation behind using classical machine learning techniques was noted to be limited availability of data.

Section 4. A discussion of goals and next steps

As previously stated, the goal of this project is to build an algorithm that analyzes an image of a Petri dish with colonies from different bacterial species and classifies them by species.



Picture 4. A sample image segmentation problem[12]

The **first** immediate goal will be to improve our initial patch extraction module. This is important for multiple reasons:

1. It allows us to use individual image patches rather than whole pictures to train the classification model, thus increasing the sample size (Data Augmentation).
2. It helps us control overfitting when training the classification model - for instance, by excluding noisy section such as the walls of the petri dish.
3. it can be used as a component for future projects related to bacteria classification on petri dishes.

The first item above is particularly relevant to successfully train neural network as deep neural networks require large amounts of data to produce accurate predictions. In addition, we can further increase the sample size by applying some basic transformation of the patches such as rotations, flipping and warping. This can be helpful to increase the size of the sample without actively having to gather more pictures of live bacteria colonies, a process that can be very time consuming and might still not be sufficient to produce a large enough sample size on its own. Although, this approach increases the risk of overfitting while training since we are not introducing novel observations but simply re-feeding manipulated images of the initial colonies to our classifier.

After improving our patch extraction module and additional Data Augmentation, our **second** goal will be building a baseline model that can work with the limited dataset we have available. For this, we will use shallow pre-trained computer vision models.

We briefly discussed the possibility of looking for additional datasets, which we could use to train a model that detects general features common to many bacterial colonies and then fine-tune it to classify the specific bacterial species for the task. After receiving feedback from project sponsors however, we came to the conclusion that finding a dataset of such size and quality might be difficult, and there is no guarantee that a model trained on different bacteria species can be helpful for our task. Therefore, we will keep this option as a lower priority task.

Finally, in case that the data augmentation is not sufficient to train a deep neural network from scratch, we plan to also use not the common pre-trained models on general image classification tasks but trying to find a model pre-trained on a similar task where the learnt features may be more useful to our problem and apply transfer learning to adapt it for our specific bacteria classification problem, training the last few layers of the deep learning network.

Contribution to the report

Andres Rios (dar2196): Sections 1, 2 and 4, plus final editing and proofing.
Akhil Punia (ap3774): Section 3. Literature Review, Report Editing and Compilation.
Carlo Provinciali (cp2984): Section 4. Discussion of Goals and Next steps
Paridhi Singh(ps3060): Section 1. Description of goal and data
Xinyuan Cao (xc2461): Section 1. Description of the problem and data collection
Zhejin Dong (zd2221): Section 2. A description of initial data exploration

References

[1] Krizhevsky, A., Sutskever, I. and Hinton, G. (2019). *ImageNet Classification with Deep Convolutional Neural Networks*. [online] Papers.nips.cc. Available at: https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks

[2] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. and Fei-Fei, L. (2019). *ImageNet Large Scale Visual Recognition Challenge*. arXiv.org. Available at: https://arxiv.org/abs/1409.0575

[3] He, K., Zhang, X., Ren, S. and Sun, J. (2019). *Deep Residual Learning for Image Recognition*. arXiv.org. Available at: https://arxiv.org/abs/1512.03385

[4] Girshick, R., Donahue, J., Darrell, T. and Malik, J. (2019). *Rich feature hierarchies for accurate object detection and semantic segmentation*. arXiv.org. Available at: https://arxiv.org/abs/1311.2524

[5] Ronneberger, O., Fischer, P. and Brox, T. (2019). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. [online] arXiv.org. Available at: https://arxiv.org/abs/1505.04597

[6] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. and Ng, A. (2019). *CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning*. arXiv.org. Available at: https://arxiv.org/abs/1711.05225

[7] Huang, G., Liu, Z., van der Maaten, L. and Weinberger, K. (2019). *Densely Connected Convolutional Networks*. arXiv.org. Available at: https://arxiv.org/abs/1608.06993

[8] Sayres, Rory et al., Using a Deep Learning Algorithm and Integrated Gradients Explanation to Assist Grading for Diabetic Retinopathy, Ophthalmology, Volume 126, Issue 4, 552 – 564

[9] Liu, Y., Jain, A., Eng, C., Way, D., Lee, K., Bui, P., Kanada, K., Marinho, G., Gallegos, J., Gabriele, S., Gupta, V., Singh, N., Natarajan, V., Hofmann-Wellenhof, R., Corrado, G., Peng, L., Webster, D., Ai, D., Huang, S., Liu, Y., Dunn, R. and Coz, D. (2019). *A deep learning system for differential diagnosis of skin diseases*. arXiv.org. Available at: https://arxiv.org/abs/1909.05382

[10] https://doi.org/10.1186/s12976-018-0093-x

[11] https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0184554

[12] Tal Danino, Lars Dietrich,et al., Machine learning-based characterization of bacteria isolates from clinical samples