BU Department of Computer Science

Classification of Bacteriophage Genome

Group 18: Alex Barrett, Kaiyu Yang, Po-Yu Hsieh, Tanmoy Pal, Theresa McNeil

Project Task

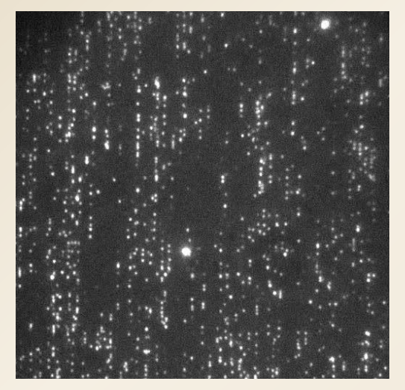


Figure 1. Genome containing enterobacteria phage λ (lambda phage).

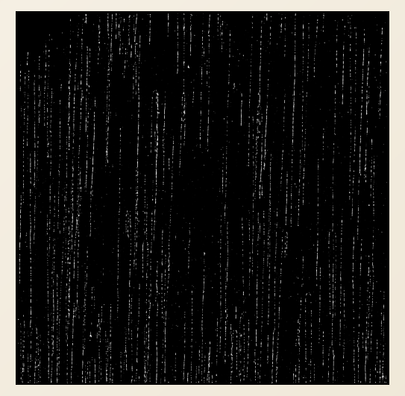


Figure 2. Aggregate image of 5000 frames of a lambda phage experiment

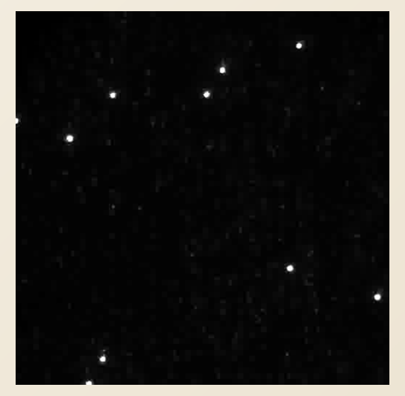


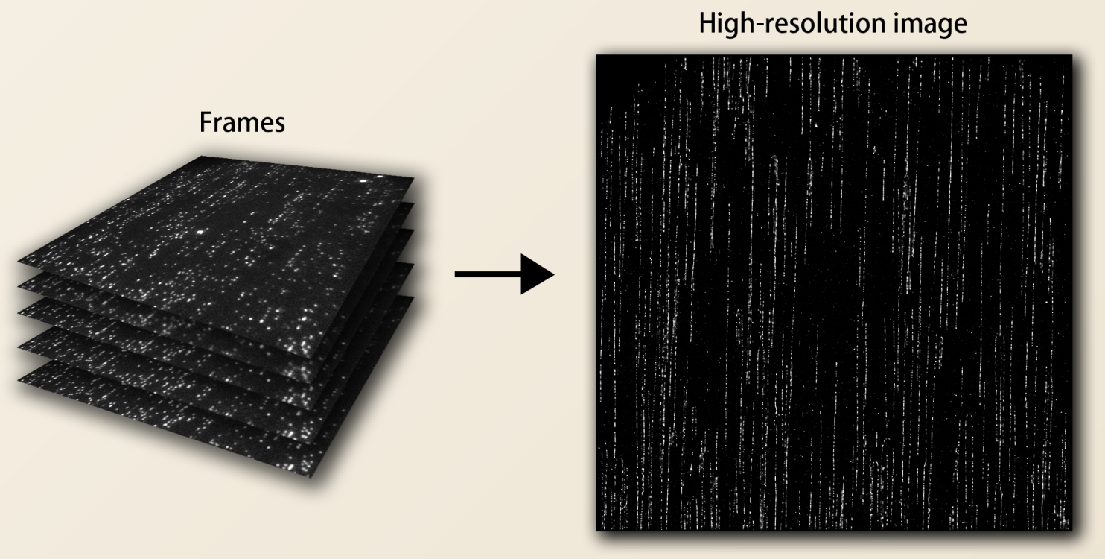
Figure 3. Genome containing bacteriophage (T7)

Figure 4. Aggregate image of 1000 frames of a T7 phage experiment

The goal of this project is to identify whether a given image of a DNA sample contains the genome of bacteriophage (T7 phage) or enterobacteria phage λ (lambda phage) based on its binding activity.

Dataset & Metrics

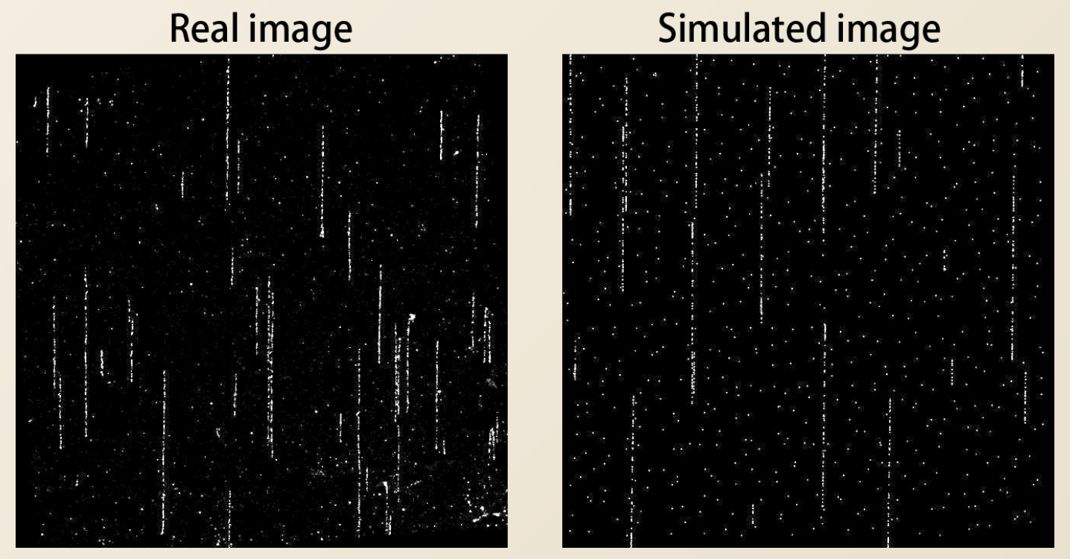
- High resolution raw movies of oligo binding activity to DNA of T7 and lambda phage from four independent experiments
- Super-resolution image output reconstructed from the movies and the localization data from the signals in csv format



	Training (lambda/ T7)	Validation (lambda/ T7)	Testing (lambda/ T7)	Total (lambda/ T7)
Frames	7064 / 5,600	1009 / 800	2018/ 1600	10092 / 8000
Molecules	555 / 327	79 / 46	218 / 93	793 / 468
Fake data	700 / 700	100 / 100	100 / 100	900 / 900

- Experiments are independent but the frames within each experiment are not, as they correspond to oligo-binding of same spatial arrangement of DNA molecule. Frames are not appropriate to be used as training data.
- DNA strands of different lengths have been identified from 8 super-resolution images using RetinaNet and used as data set.

- Simulating the binding behavior in ideal experiment, “fake” data was generated using NEBcutter with adjustable parameters such as the density levels of the oligo, and the genome.

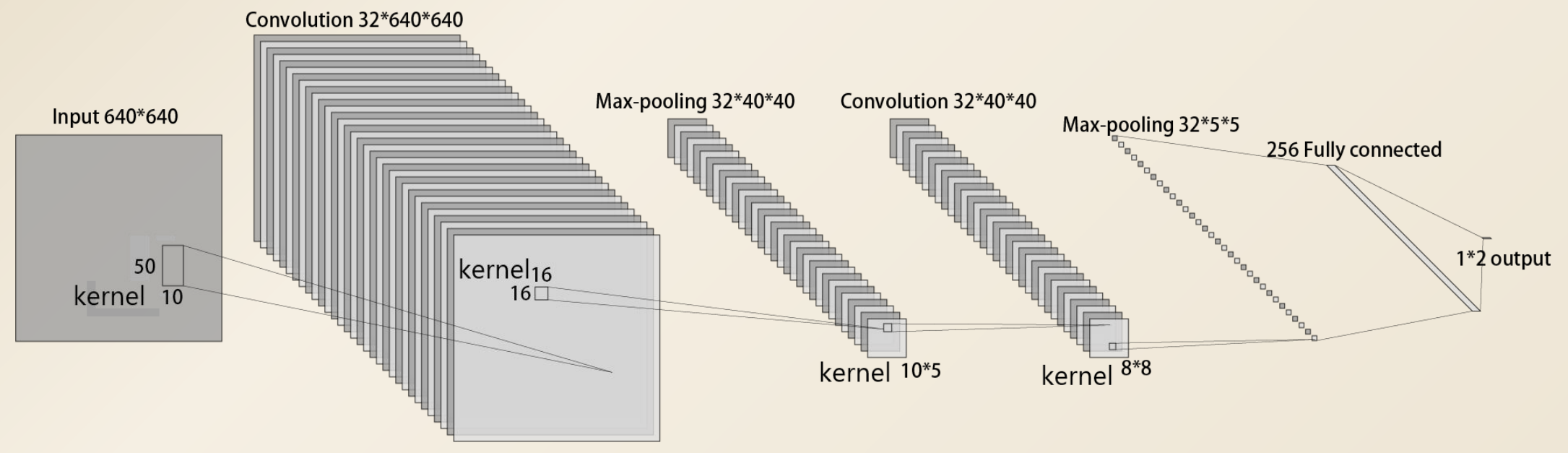


- Success evaluated by F1 score, precision & recall.

$$F_1 = \left(\frac{\text{recall}^{-1} + \text{precision}^{-1}}{2} \right)^{-1}$$

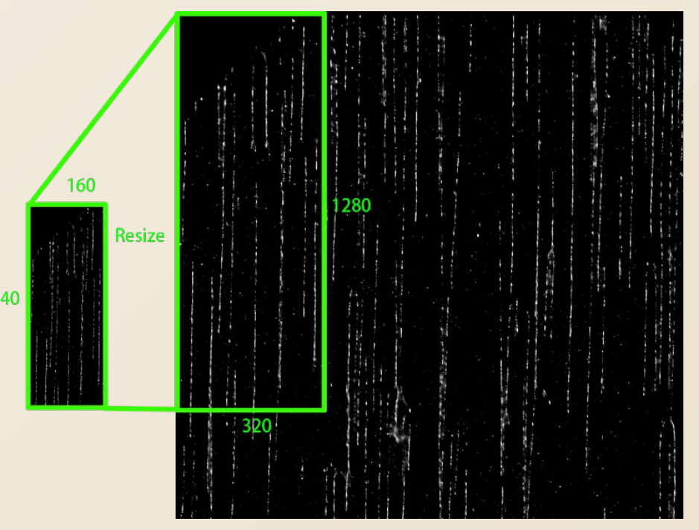
Methods

Approach 1. Train CNN on individual frames of video.

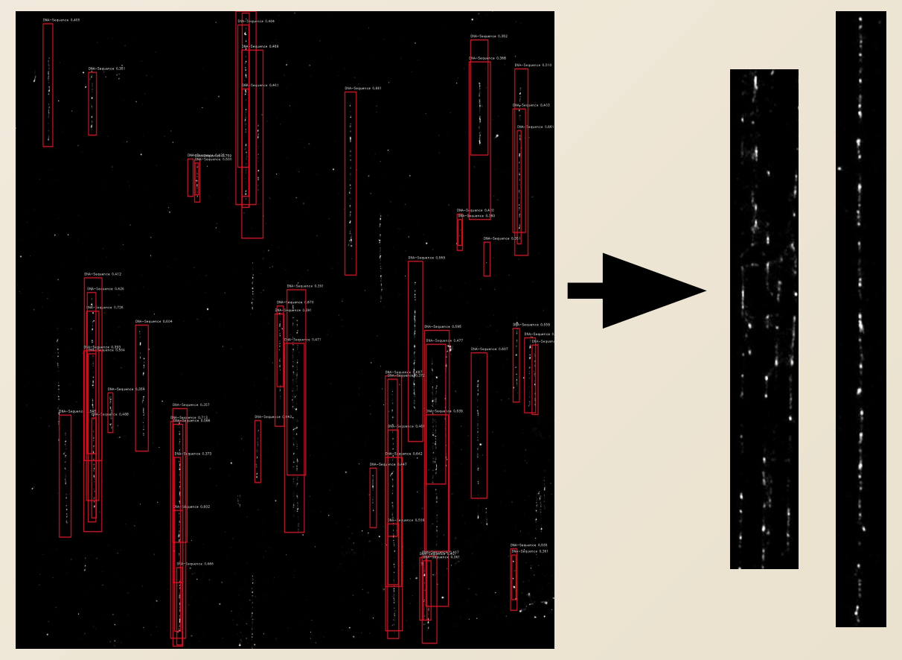


Approach 2. Train CNN on our simulated images and test it on real aggregate images

Approach 3. Cut aggregated image into small pieces with a uniform sliding window and train CNN on them



Approach 4. Train CNN on extracted molecule strands



DNA strand detection from super-resolution images using RetinaNet object detection network

Results

1. CNN based on frames is not successful due to little difference between each frame. Produce performance no better than random.
2.

	adjusted distribution	adjusted density	adjusted molecule size & noise	adjusted noise	real data
accuracy	94.00%	99.50%	91.50%	95.00%	26.0%
recall	88.46%	99.00%	100.00%	100.00%	0.00%
precision	100.00%	100.00%	85.55%	90.96%	0.00%
f1	93.88%	99.49%	92.91%	95.27	0.00%
Random f1	50.00%	50.00%	50.00%	50.00%	50.00%

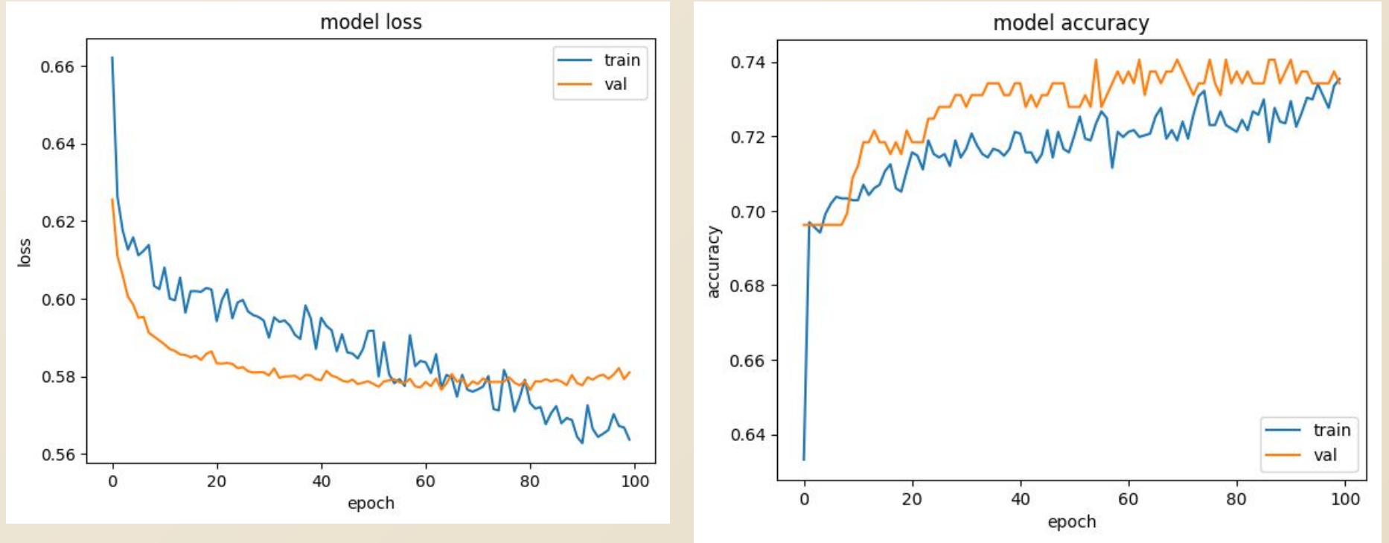
Good results on simulated data, which immune to the variance of molecule position, density and noise occur in simulated data; does not generalize well to real data

3.

	test on 3rd experiment	test on 4th experiment
accuracy	90.2%	43.6%
recall	84.77%	94.31%
precision	92.35%	44.21%
f1	88.40%	59.95%
random f1	50.00%	50.00%

In the first 3 experiments the lambda molecules were very dense and the T7 were very sparse but both 4th experiments contained similarly dense molecules. The model learned to falsely classify based on the densities.

4. Density differences also caused a large imbalance of molecules extracted for each class, & accuracy close to random, but other metrics prove model did learn.



accuracy	70.7%
recall	94.1%
precision	72.4%
f1	82.1%
random f1	58.43%

Conclusion

The biggest challenge we faced was figuring out an effective way to use the data we were provided. Given our success with the model that classified fake data with high success, we may have been able to achieve similar success with more experiments. If we could generate our own experiments we would be sure to include a varying density of molecules for both classes to prevent the model from learning based on density, as seen in the results of method 3.

@BUCompSci