

# Are Those Ants?

**Presenter: Karanveer Gill · Computer Information Systems · Mentor: Dr. Sarbani Banerjee**

## Abstract

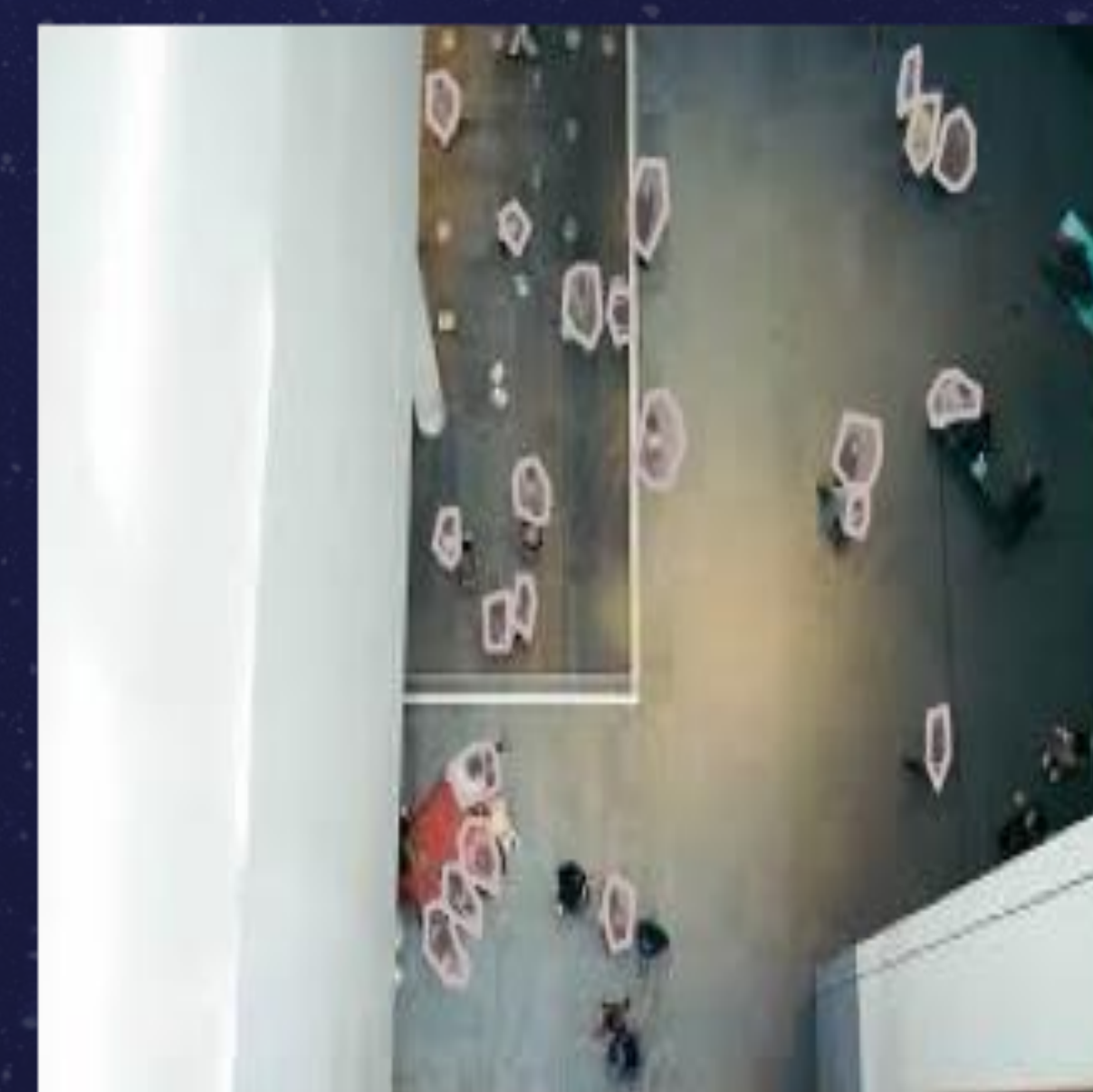
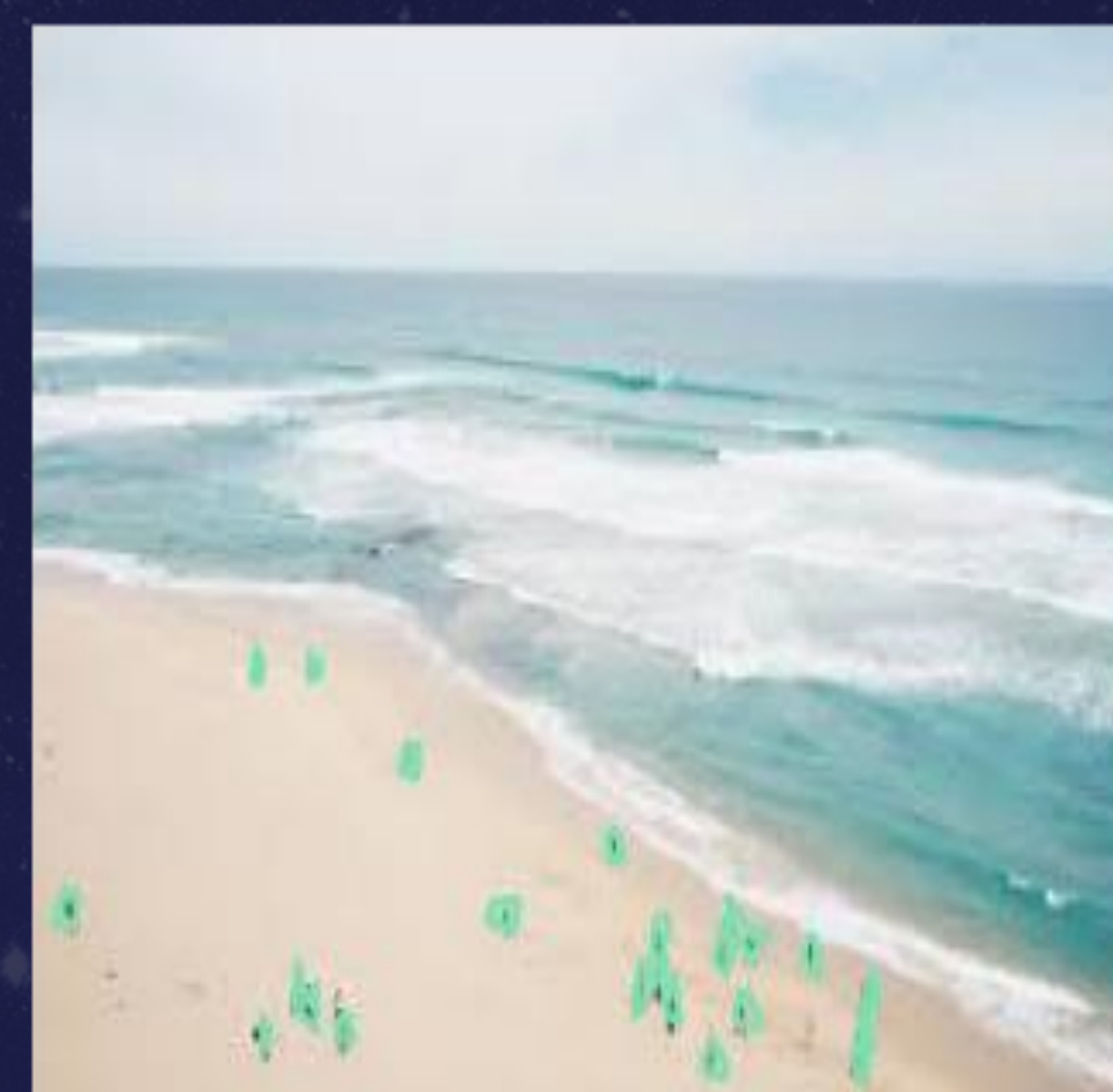
This project will be studying how to detect people in images who appear miniscule. Small people can be identified as low as 20 pixels. This project has some inspiration from satellite images. It can almost replicate how a satellite image can detect a person. For example, if a picture was taken from a satellite, the identification of small persons will only be beneficial for investigations. This project will be written by using Python language and Jupyter Notebook will be used as the Integrated Development Environment (IDE). Windows Docker will be utilized to access a COCO annotation. This will be accessed within the Python program, referencing a JSON file. A dataset will be downloaded with help from this COCO annotator. This data annotation will be designed to detect tiny people. The purpose of this project is to immediately detect persons in a picture that are not in a crowded area. Images will mainly be aerial shots where the people in the image will look tiny in size. There will be an indication that the object, in this case the “tiny-person,” is identified. This project will also attempt to zoom-in the photo, almost to replicate a zoomed-in camera or satellite image. With the identification of tiny persons, the object detected will turn out to be human. An objective of this research is to mimic a satellite image or camera identifying potential suspects.

## References

GitHub Repository: <https://github.com/ucas-vg/TinyBenchmark>  
 Tiny Person Detection: <https://arxiv.org/abs/1912.10664>  
 PyCharm: <https://www.jetbrains.com/pycharm/download/>  
 CoCo Dataset: <https://cocodataset.org/#home>  
 CoCo Annotator: <https://github.com/jsbrooks/coco-annotator>

## Methodology

The project *Are Those Ants* requires a few tools. First, a COCO Annotator will be utilized to manually annotate the object that is going to be detected. In this case, people. Once uploaded, the annotations dataset will be available as a JSON file. Secondly, Jupyter Notebook will be accessed to upload the downloaded dataset and to import images. The program’s code will be tailored to start the segmentation process, which will also include an indicator around the person, signifying the successful detection.



## Interpretation of Findings

From my findings, I have learned that successful image segmentation requires certain IDs to successfully work. Segmentation and identification go together with datasets which can be customized using a custom dataset. You can create a custom dataset using a COCO annotator, downloading a pre-made one or using a program like LabelMe. Segmentation occurs by extracting the category and image ids. By doing this, the program you create can look for the category in the image.

## Conclusion

In conclusion, I was able to use a COCO annotator to create a custom dataset. This dataset was used to create a program to successfully perform image segmentation on a specific category, in this case the category was “person”. Although, I was able to successfully create an image segmentation program, I was not able to create incorporate a zoom feature. This zoom feature’s intentions were to replicate a zoomed-in image of a camera or satellite. However, I have been told that a project like this would be too difficult for a student with 0 knowledge in machine learning. Through, extensive research I was at least able to accomplish the main goal of image segmentation. If granted more time with even more research than before, I have no doubt I will be able to accomplish the previous goals I have set.

## Acknowledgements

I would like to acknowledge the Computer Information Systems Department at Buffalo State College and my Advisor, Dr. Sarbani Banerjee. With their continued support, I was able to complete my project.