2022-12-01

Oil Particle Analysis Using Machine Learning and Holography Imaging

Daniel Cruz
University of Texas at El Paso

OIL PARTICLE ANALYSIS USING MACHINE LEARNING

AND HOLOGRPAHY IMAGING

DANIEL CRUZ

Master's Program in Computer Science

APPROVED:

Olac Fuentes, Ph.D., Chair

Craig Tweedie, Ph.D.

Diego Aguirre, Ph.D.

Stephen L. Crites, Jr., Ph.D.
Dean of the Graduate School

Dedication

I dedicate this work to all my friends and family that cheered me on to finish this chapter of my life. Especially mom and dad, thank you for always providing me a place to grow and pursue myself.

OIL PARTICLE ANALYSIS USING MACHINE LEARNING AND HOLOGRPAHY IMAGING

by

DANIEL CRUZ

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Computer Science

THE UNIVERSITY OF TEXAS AT EL PASO

December 2022

Acknowledgements

I'd like to thank Gesuri Ramirez and Dr. Craig Tweedie for being great bosses and mentors in my professional life. I would not have had the opportunity I have now without the mentorship from Dr. Rodrigo Romero in my undergraduate years. Its through his guidance that I was able to achieve my academic goals. Thank you Dr. Lisa DiPinto, Amy Kukulya, Dr. Nancy Kinner, and everyone I met and collaborated with on my summer internship. This research would not have been possible had it not been for the kind and gracious mentors and scientists I met on my internship at scibotics lab and Coastal Response Research Center. This work would not have been completed had it not been for my graduate advisor Dr. Olac Fuentes' and Vision and Learning Lab's diligent guidance and patience throughout my academic journey. Lastly, I would like to thank Dr. Diego Aguirre for his continued support as my committee member.

This research was supported through NOAA Educational Partnership Program/Minority-Serving Institutions award number NA16SEC4810008 to the Center for Earth System Sciences and Remote Sensing Technologies. Contents are solely the responsibility of the author(s) and may not represent official views of NOAA or the U.S. Department of Commerce. The authors would also like to thank NOAA Office of Response and Restoration, Woods Hole Oceanographic Institution, Scibotics Lab, and University of New Hampshire Coastal Response Research Center for hosting and assisting in this research.

Abstract

Holographic cameras show potential as a sensor to monitor oil spills. Holographic cameras record the light interference from particles in a volume of space, producing an image called a hologram. Processing these holograms is known as hologram reconstruction. It produces a representation of particles located in three-dimensional space. These cameras can record precise shapes and sizes of particles in a volume of water. However, it is very time-consuming and resource-intensive to process the images. Most algorithms that perform particle analysis require the hologram reconstruction step. The well-documented hybrid method is one such algorithm. Machine learning is one possible technique that shows potential for avoiding the costly hologram reconstruction altogether while extracting particle statistics. We tested two different machine learning models, U-Net with a resnet34 backbone and YOLOv5, and compared their performance. We collected a dataset of real-world hologram images of oil particles. We then trained our machine learning models on this real-world data with hybrid method derived labels. We used average precision and average recall to compare model performance. Compared to the hybrid method, we demonstrated a processing time reduction of 99%. YOLOv5 with GPU acceleration proved to be the superior model in all metrics. Machine learning is a promising technique for processing these hologram images rapidly. This allows quick-response agencies such as oil spill response teams the ability to use holographic cameras to assess oil spills.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The general perception of an oil spill in the ocean is viewed as a continuous rainbow sheen floating on the surface of the water. This view however oversimplifies what is occurring below the surface of the ocean. Mechanical forces of an oil spill blowout and chemical dispersants cause oil to take on the form of tiny droplets underwater [1]. These droplets can range anywhere in the nanometer to micrometer range. This makes them very difficult to track and analyze due to their small size and distribution in the ocean. This makes monitoring an oil spill a significantly difficult task. Analyzing oil droplets is part of an assessment of an oil spill [2]. An emerging practice is to use autonomous underwater vehicles, as shown in figure 1.1a, to assess oil spills underwater. This allows quick-response teams to assess the state of the spill quickly and safely. These vehicles can stay underwater for weeks and report readings from the attached sensors in real-time. Scientists often utilize a whole suite of sensors to locate and analyze these droplets. The sensors include flourometers, water quality sondes, and water samplers [2]. The holographic camera sensor shows great promise as a tool to analyze oil droplets.

Holographic cameras, particularly in-line holographic cameras, have a laser emitter aligned with a lens-less camera tuned to the wavelength of the laser. Figure 1.1b shows what a holographic camera typically looks like. Particles in the volume between the laser emitter and the camera collide with the light and the light is scattered. Instead of capturing the object itself, the camera records the scattered light in the form of a diffraction pattern. Figure 1.3a depicts this process. We can then direct a laser with the same power and wavelength as the recording laser at the pattern. This pattern will then diffract the light as it passes through it. If we place a background in the path of the beam, we can adjust

(a) (b)

Figure 1.1: (a) Holographic camera mounted in front of an autonomous underwater vehicle designed for monitoring oil spills. (b) Holographic camera mounted in a metal harness.

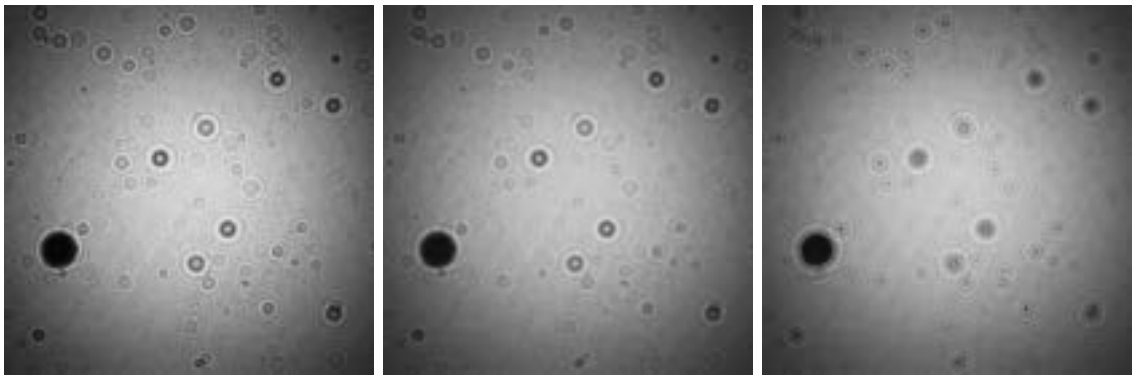
the distance away from the diffraction pattern in such a way that the diffracted light will recreate the shape and size of the particle. The distance between the pattern and the reconstructed particle is the same as the distance the particle was from the camera. Figure 1.3b depicts this process.

Originally this was done with laser experiments. However, modern-day computing power affords the ability to computationally simulate this process. This is known as reconstructing the hologram through numerical propagation [3]. When the hologram images are processed, scientists can analyze a three-dimensional reconstruction of the volume of particles between the laser and the camera. This reconstruction process produces images that represent slices of different planar locations in the original captured volume. By moving through these slices, we can see diffraction patterns converging until we reach the slice of the particles' placement in the volume. Figure 1.2 provides a visual example of different reconstructed slices. This reconstruction not only captures the three-dimensional location of a particle but also accurately captures its shape and size [4].

This sensor demonstrates several useful features for analyzing oil droplets. Due to the speed of the imaging system, the images contain very few motion artifacts for moving objects. A single image records data that can rebuild particle statistics in a volume of water. Because of this, it is not necessary to focus the camera on a single focal plane before data capture. We can focus the image after capturing the data and we can analyze several particles in one image.

Holographic cameras have a major disadvantage: difficulty in analyzing the images. The reconstruction process is time-consuming and computationally expensive. Because the reconstruction is a 3D representation, conventional computer vision techniques will not work unless adapted specifically for holography. Holographic machine learning techniques have made great strides in the field of computer vision. Many machine learning applications for holographic images produced successful results [5, 6]. Researchers have demonstrated that machine learning can circumvent the need to perform the costly reconstruction process.

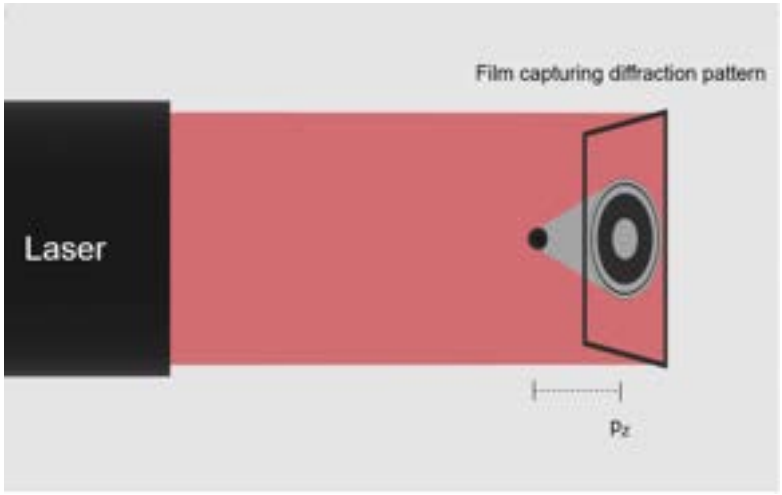
In this work, we applied deep learning to holographic imaging to rapidly analyze oil



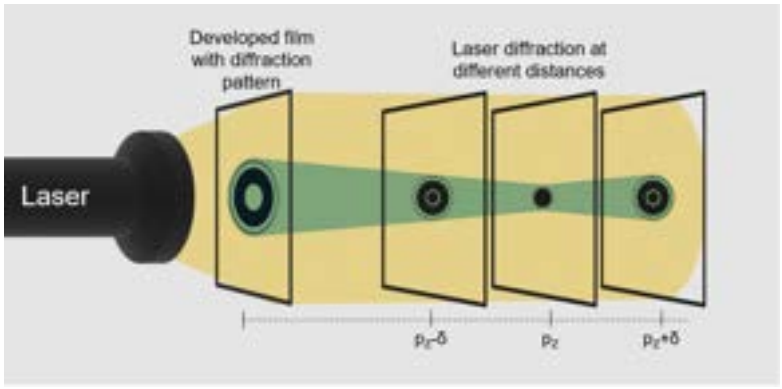
(a) (b) (c)

Figure 1.2: (a). Raw hologram. This is what is recorded on the camera, or at a depth of 0 mm from the camera. (b). A reconstruction slice of the hologram at depth 23 mm from the camera. (c). A reconstruction slice of the hologram at depth 85 mm from the camera. In each slice, particles that were diffraction patterns in the other slices come into focus.

particles in the images. We focused primarily on statistics of interest to oil spill assessment, particularly particle count and particle size. The value of this project is a rapid method to extract data from hologram images. Oil response teams can use this method to rapidly monitor oil spills in the ocean using holographic cameras.



(a) Hologram recording using in-line holography. A particle is a distance p_z away from the camera. The particle collides with the light and the camera records the resulting diffraction pattern. The camera records a hologram.



(b) Optical hologram reconstruction. A laser with the same characteristics as the recording laser is shown through the diffraction pattern on the hologram. A film plate in the direction of the beam, when positioned at the correct depth, will record a reconstruction of the particle that created the diffraction pattern. This correct depth is the distance p_z away from the diffraction pattern.

Figure 1.3: Recording and reconstructing a hologram optically. While digital holography can perform both processes numerically, the concepts remain the same.

Chapter 2

Related Work

2.1 Algorithm for Reconstructing Holograms

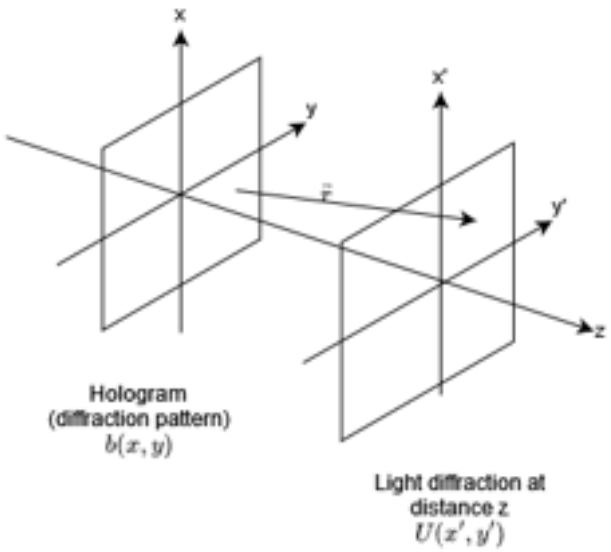


Figure 2.1: Reference geometry of hologram reconstruction.

In this section, we will use figure 2.1 as a reference for geometry. Diffraction patterns in the hologram image produce all the information necessary to recreate a 3D volume of particles. Each diffraction pattern in the hologram represents a particle between the laser and the camera. Reconstructing the particles involves calculating how light is diffracting through the patterns. In the early days of optical holography research, researchers would develop the diffraction patterns on film and pass the laser through the film to view the reconstruction [7].

Currently, digital holography performs this process computationally using light models.

One of the first articles on calculating this process numerically is [8], where they list the equations they used to model the reconstruction. The way light diffraction is modeled depends on the size ratio between the distance of the camera and the wavelength of the light [4,9]. Referencing figure 2.2, holographic cameras can operate in the Rayleigh-Sommerfeld & Fresnel-Kirchoff or Fresnel (near field) regime. This is due to the much greater distance between the camera and the laser when compared to the laser's wavelength. In this regime, the diffraction of the light is calculated with the Rayleigh-Sommerfeld diffraction equation:

$$U(x', y') = \frac{1}{i\lambda} \int \int b(x, y) \frac{\exp(ikr)}{r} dxdy$$

where:

$$r = \sqrt{(x - x')^2 + (y - y')^2 + z^2}$$

$$k = \frac{2\pi}{\lambda}$$

In this equation $b(x, y)$ is the diffraction pattern, λ is the wavelength of the laser, i is the imaginary unit, and k is the wavenumber. Note that because this equation computes a complex amplitude, we will be able to calculate both the amplitude and phase of light.

We can take two routes of calculations in order to simplify the computation. One route is to assume z the distance between the hologram and diffraction plane is large compared to $(x - x')$ and $(y - y')$. When this is the case we can use the Fresnel approximation.

However, we shall use the second route for reasons that will be explained below. We may consider the Rayleigh-Sommerfeld diffraction formula as:

$$U(x', y') = \int \int b(x, y) g(x, y, x', y') dxdy$$

where:

$$g(x, y, x', y') = \frac{1}{i\lambda} \frac{\exp ik\sqrt{z^2 + (x - x')^2 + (y - y')^2}}{\sqrt{z^2 + (x - x')^2 + (y - y')^2}}$$

We can write the function $g(x, y, x', y')$ as the form $g(x - x', y - y')$. From here we see that:

$$U(x', y') = \int \int b(x, y) g(x - x', y - y') dxdy = b(x, y) \circledast g(x, y)$$

$U(x', y')$ is a convolution, denoted by $\circledast$, of functions b and g. Using the Fast Fourier Transform we can calculate this convolution as a product in the frequency domain. Then we can use the reverse Fast Fourier Transform to calculate the diffraction of the light:

$$U(x', y') = \mathcal{F}^{-1}(\mathcal{F}(b) * \mathcal{F}(g))$$

There is a distinct advantage to using this method of calculation as opposed to the Fresnel approximation. While processing at similar times, this route guarantees each reconstructed slice will maintain the same resolution per pixel as the raw hologram image [10]. This is a very important utility for measuring particles as they appear in the reconstructed slices. We can use this process to reconstruct image slices at determined intervals that correspond to a set of planes between the laser and the camera. From here on we shall refer to this set of reconstructed images as the stack, and each image as a slice.

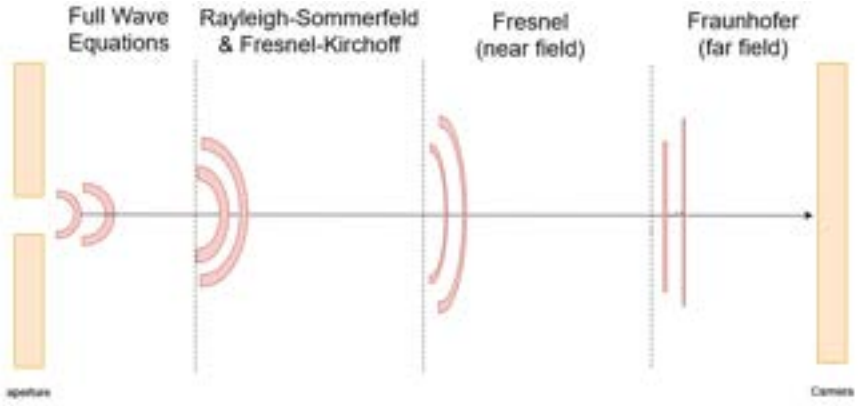


Figure 2.2: Diffraction Regimes. The farther light travels from the aperture, the more light can be modeled as a scalar rather than a spherical wave vector.

Another method exists to recreate the 3D volume that the hologram captures. This method is the inverse-problem approach. Just as we can model light diffraction using the light regime models, we can also model light interference from particle clouds. The inverse-problem approach models the holographic camera setup and seeks to find a set of particle locations and particle sizes that optimize the similarity to the raw hologram [11]. This method offers a faster approximation since the hologram does not have to be reconstructed.

It is also not biased toward particles near the edge of the hologram. These particles may have their diffraction patterns cut off on the edge of the hologram, making the hologram reconstruction less accurate. The limitation of this method is that the execution time now scales with the number of particles in the hologram image. The experiments conducted in [11] demonstrate approximately 7 minutes per particle.

2.2 Non-Machine Learning Methods to Analyze Particles

One of the very first applications for holography imaging was particle analysis [12]. One early method uses the minimum pixel intensity of the reconstructed slices to find the slice with the particle in focus [9]. The experiment involved synthetic holograms with a single particle. The team used minimum pixel intensity to find the slice with the particle in focus. Then they used a threshold to extract the particle and measured the diameter of the particle. They were able to calculate the diameter of the particle in the synthetic hologram with a precision of 2 significant figures. This work proved the ability to measure particles with holographic cameras. However, many of the experiments only involved analyzing a single particle in the hologram image. Measuring oil particles with a holographic camera often means recording many particles.

Later research included methods that involved measuring and analyzing the sharpness of particle edges and developing a sharpness criterion. This method maximizes the sharpness of the image. In [13], the authors tested several edge sharpness quantifying algorithms to analyze the volume of red blood cells. The researchers placed the red blood cells on a glass slide. This represented a single-plane target where all the red blood cells were in focus. The algorithms found the slice with the best sharpness. The sharpest slice had all the particles in focus. They then compared the output from each depth algorithm and used a bandpass filter on the image intensity to segment the red blood cells based on a set

threshold. The best algorithm had an uncertainty value of tracking a red blood cells ±0.3 μm in the x,y location and with an uncertainty of ± 2.4 μm. Unlike the previous method in [9], these experiments involved holograms that captured multiple objects. While the red blood cells all existed in a single plane of reconstruction, the experiments demonstrated that holography imaging is useful for analyzing a collection of objects.

Most hologram analysis methods require reconstructing the hologram. Hologram reconstruction is the only way to visibly see the particles encoded in the hologram. [14] presented a method to perform particle analysis without hologram reconstruction. Trading precision for quick analysis, the authors demonstrated a method to measure the mean diameter of particles based solely on the raw hologram image. The method focused on the diffraction patterns on the hologram and how the particle data is encoded. The center of the pattern is the center of a particle, the frequency modulation of the concentric rings encodes the depth of the particle, and the amplitude modulation is correlated to a distance-to-diameter ratio. On synthetic holograms, this method was able to estimate particle mean diameter with an absolute error of less than 5 μm.

Gao and Guildenbecher *et. al* [15–17] introduced the hybrid method, named as such because it combines concepts from the two previous techniques. The hybrid method is an algorithm designed to segment particle clouds in a hologram image. First, the algorithms reconstruct the hologram. Then as in [9], the algorithm builds a minimum intensity map between each slice on the stack. Then as in [13] the algorithm uses an edge sharpness criterion based on Sobel edge detection. The algorithm uses the edge sharpness criterion to test a range of thresholds applied to the minimum intensity map. The threshold that maintains the sharpest edges produces a binary particle mask. The algorithm also outputs a depth map that estimates the depth of each particle in the particle mask. When tested on synthetic holograms, the hybrid method detected approximately 90% of the particles in a hologram, with an error percentage less than 1% in both location and depth. An experiment with real-world holograms proved that the hybrid method outperformed other methods, demonstrating the hybrid method's robustness towards noise. Gao *et. al* later

improved the hybrid method by using a depth map to differentiate occluding particles in the particle mask. A downside for this algorithm is the high computational expense, taking several minutes and large amounts of memory to run on a single hologram. This makes it impractical to use this method with a holographic camera as a quick response sensor.

2.3 Machine Learning Methods to Analyze Particles

In recent years there have been several successful applications of machine learning in the realm of particle analysis in holographic imaging. Schneider *et. al* [5] demonstrated one of the first applications of neural networks to holography. This experiment focused on a single coated particle in a hologram. A coated particle has an inner diameter and an outer translucent diameter. A simple densely connected model received this raw hologram and predicted the particle size. The pixels of a raw hologram was the input. The outputs were the particle's outer diameter and inner diameter. Although a very simple model for a computer vision problem by modern standards, this model was able to predict particle diameters with a root mean square error of 4.4. Schneider proved that neural networks can provide accurate predictions while also avoiding the costly reconstruction process.

Shimobaba *et. al* [18] developed a model to predict the 3D location and size of a particle distribution. The team designed a U-Net model with 256x256 hologram patches as the input and three map outputs: x-y location map, depth map, and size map. The team trained the model on a dataset of synthetic hologram images. To test the model's robustness against background noise, all the images included Gaussian noise. The results showed that the model predicted location with an average error of approximately 0.3, depth of approximately 3.5, and size of approximately 4.0. It took approximately 0.3 seconds to process a hologram image.

Shao *et. al* [19] used a deep neural network, a U-Net, to analyze particle distributions. The inputs for the network were a raw hologram, a slice, and a minimum intensity map similar to that used in the hybrid method [16]. The outputs were a particle size map and

an x-y location map. This model required each slice of a stack to predict all the particles in the hologram. The team's workflow required the costly reconstruction process when using the model. This workflow had slower execution time but higher precision. Trained and tested on synthetic holograms, the model had a 91% detection rate and an x-y positional error less than a particle's diameter. The experiments included a test with spherical and elliptical particles that produced similar results. This showed that this model was not limited to spherical particles only. A test with real-world hologram images demonstrated a detection rate of 95% and a depth error of less than a particle diameter.

Valadares *et. al* [20] used a U-Net and an auto-encoder neural network to segment spherical particles in simulated hologram images. The input for both models was a single reconstructed slice. The team generated these synthetic holograms and added a background of a real-world hologram with no particles. This added realistic noise. The U-Net outperformed the auto-encoder with the best network having an object error of 4.75%. This work used the inverse-problem approach to generate masks for real data. The masks provided a way to visually compare the models' predictions with real-world hologram images. The authors concluded that while the machine learning approach was not as accurate as currently established methods, it could be used as a quick estimation.

[21] tested a holographic camera as a real-time sensor. They constructed a small autonomous underwater vehicle with a holographic camera. For rapid data extraction from the camera, they trained a simple convolutional neural network with raw hologram input and air bubble concentration output. The experiment demonstrated the robot mapping air bubble concentration in real-time in a laboratory experiment and a field experiment in a lake. This work shows the applicability of holographic imaging and machine learning as a quick response system.

Currently, there has been interest in object detection models for holographic imaging. [22] showcased a YOLO network's performance in detecting particles in raw holograms. They used the OSNet architecture. The team generated synthetic holograms. They modified the loss function of the network to include depth information. The model learned the

full 3D coordinates of a particle. The model had a 98% recall and 99% precision on the simulated dataset.

Chapter 3

Approach to Speed Up Hologram Analysis

3.1 Data

An important aspect of any machine learning approach is a large and varied dataset. This dataset should closely represent real-world data. To this end, we set up experiments to collect holographic images with oil droplets. We used a Seascan Inc. holographic camera for the experiments. This camera captures a 1.12 cm by 1.12 cm by 15.9 cm column of water. The images produced are 2048x2048 pixels with a resolution of 5.5 µm per pixel. We established two experimental setups. Figure 3.1 shows these two experimental setups.

(a) (b)

Figure 3.1: (a) Small-scale experiment (b) Large-scale experiment

First we established a small-scale experiment designed to quickly generate holograms of oil droplets. We submerged the sensor underwater with a magnetic stirrer. We then

injected oil close to the stirrer using a syringe. The stirrer broke down the oil into oil droplets that the camera then recorded. We used MC20 oil for these experiments with datasets of varying concentrations of oil droplets.

Our second experiment was a large-scale experiment where we generated datasets with varied oil types and experimented with dispersant chemicals. We submerged the holographic camera in a circulating flume that simulated underwater currents. We injected MC20, ANS, and Hoops oil into the flume and collected holograms of the oil droplets. We experimented with corexit, a chemical used as an oil dispersant. We injected the dispersant at a ratio of 1:20 dispersant to oil. Dispersant causes oil to break down further, thus the oil droplets become smaller [1].

3.2 Machine Learning Model

The most computationally expensive component of processing a hologram image is the 3D reconstruction process. Table 3.1 shows the execution time for the different components of the hybrid method. This is a required component for most hologram processing techniques. We investigated a technique to remove the reconstruction process. We trained a machine learning model to learn a relationship between the diffraction patterns and a particle's size and location. This was done with the goal to speed up the hologram analysis process by completely circumventing the reconstruction process. We investigated the following two models:

Table 3.1: Hybrid method execution time breakdown

Hybrid Method Component	Execution Time Percentage
Reconstruction	44%
Minimum Intensity Map	1%
Edge Map	15 %
Optimal Edge Threshold Search	40%

3.2.1 U-Net

Several publications demonstrated the application of U-Net models to hologram images with impressive results [5, 6, 19, 20]. U-Nets are commonly used for object segmentation tasks. U-Net can learn how to predict a particle segmentation map, making it possible to perform particle analysis on these predicted maps. This is a simple translation from the hybrid method since the hybrid method also outputs a particle segmentation map.

Machine learning models require an equation known as a loss function. This loss function is the primary metric that a machine learning model uses to learn. When the model is converging, the loss function should decrease in value. This usually signifies better predictions. The Dice Loss is a common loss function in medical deep-learning segmentation problems [23]. This loss function is well suited in problems where the background pixels outnumber the pixels of the object of interest. We chose the Dice Loss because of the class imbalance between the background pixel class and the particle pixel class. We used a resnet34 architecture as the backbone. We used the Segmentation Models python library with TensorFlow to implement this model [24].

$$Dice\,loss = \frac{2|X \cap Y|}{|X| + |Y|}$$

3.2.2 YOLO

Although not as common as the U-Net model, YOLO models also demonstrate utility for hologram image analysis [22, 25]. YOLO models are object localization models. They classify the objects in an image and draw a bounding box at the location of the object. This model can localize particles in the raw hologram image. After such, we can use the bounding box for particle analysis. The number of bounding boxes represents the number of particles in the image. We can use the size of the bounding box to estimate the size of the particle. This is a unique application of YOLO models. These models are designed to detect objects in an image. However, the particles themselves do not show up in the raw

hologram image. Due to this, the model will predict the size and location of the particle solely based on the diffraction patterns.

We used a YOLOv5 model with YOLOv5x pretrained weights. We used the default loss function for YOLOv5, Binary Cross-Entropy with Logits Loss (BCElogit). The Binary Cross-Entropy Loss Function is commonly used for classification. This function is then combined with a logit function:

$$BCElogit = -\frac{1}{N}\sum_{i=1}^{N} y_i * \log(p(y_i)) + (1 - y_i) * log(1 - p(y_i))$$

Where y is the label, $p(y)$ is the predicted probability of a specific class, and N is the number of samples in a batch size.

3.3 Label Generation

Machine learning models require a label corresponding to the data. This label determines what the model should learn from the data. It would have taken a significant amount of effort to manually label oil particles by hand. Instead, we collected particle statistics automatically using the hybrid method. Figure 3.2 outlines the hybrid method. This method produced a particle mask that we used to train the U-Net model. Figure 3.3 shows an example of a particle mask resulting from the hybrid method. We used connected components on the hybrid method mask to then calculate the diameter and location of each particle in the mask. We formatted the bounding boxes from the output of connected components to train the YOLO network. While the hybrid method also calculates the depth of the particles, we ignored these values. Our interest was to locate the particles on the raw hologram for counting rather than finding the exact 3D location in the volume.

There was a disadvantage to the label generation workflow. Our method of labeling the data was not 100% accurate in both locating particles and extracting the size of the particles. Many related experiments used synthetic holograms for training and testing. This was a simpler method for generating holograms and ground truth labels. Our workflow used

the hybrid method, which [15] showed a 90% particle detection rate in most cases. We used a large quantity of labeled data in order to mitigate the fact that the labels were not 100% accurate.

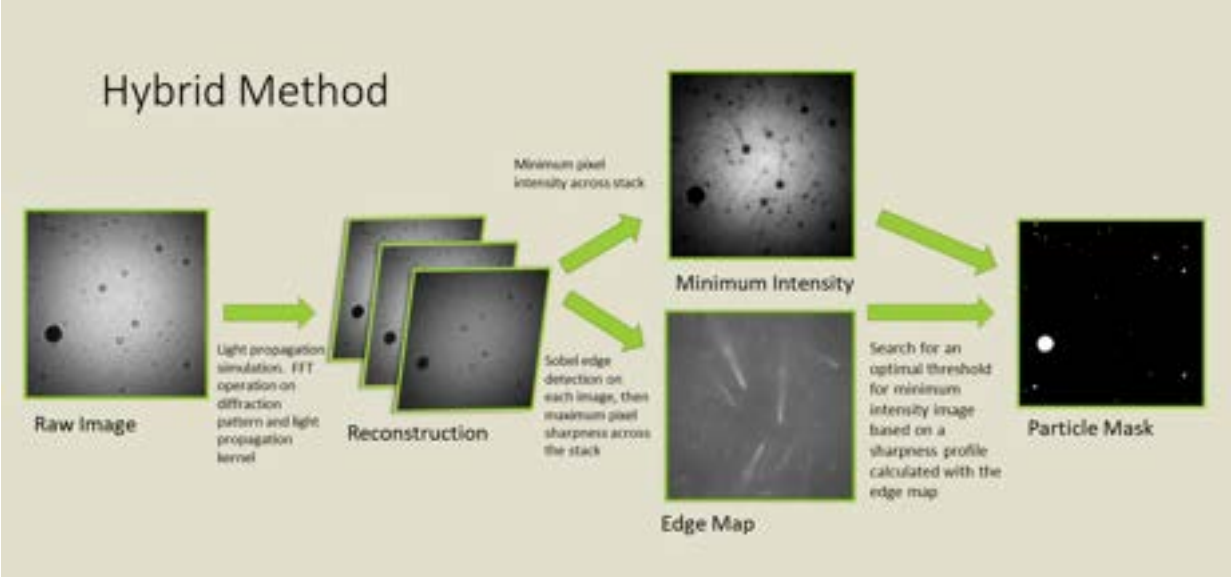
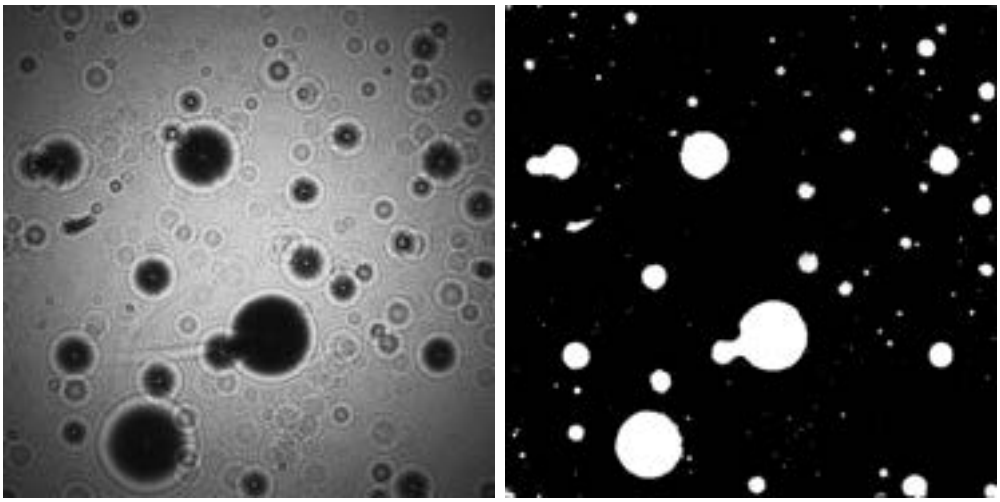


Figure 3.2: Hybrid method



(a) (b)

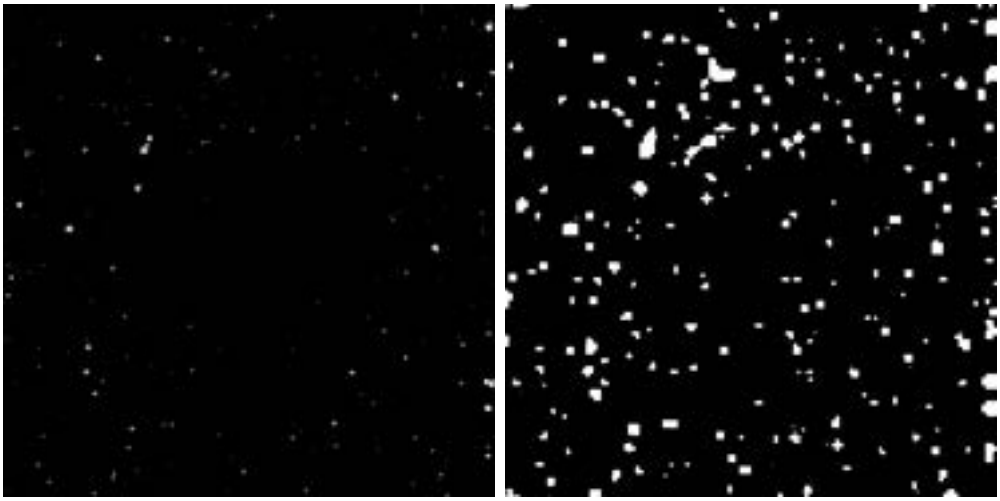
Figure 3.3: (a) Raw hologram image (b) Resulting segmentation label from hybrid method

3.4 Preprocessing

The input of the model was a raw hologram image. The original size of the images were 1024x1024 pixels. This size was too large to load into GPU memory efficiently. Initially we experimented with downsampling the images to 254x254. As noted in figure 3.4, downsampling the images lost the details necessary to reconstruct the smaller particles. The model's predictions were low in fidelity.

Instead of downsampling the images, we cropped the images. Similar to [18], we elected to crop the images into smaller 512x512 portions. For the U-Net model, we chose to crop these images with a 50% overlap. After predicting each cropped image, we stitched the images together. We averaged the overlapped regions between cropped images. We set a threshold where any pixel value at or above 0.5 is a 1, and any value less than 0.5 is a 0. The purpose of the overlap is to allow the model more opportunities to predict particles where the diffraction patterns may be cut off at the edges of a cropped image. Overlapping the cropped images created a higher chance that an image would contain the majority of a diffraction pattern.

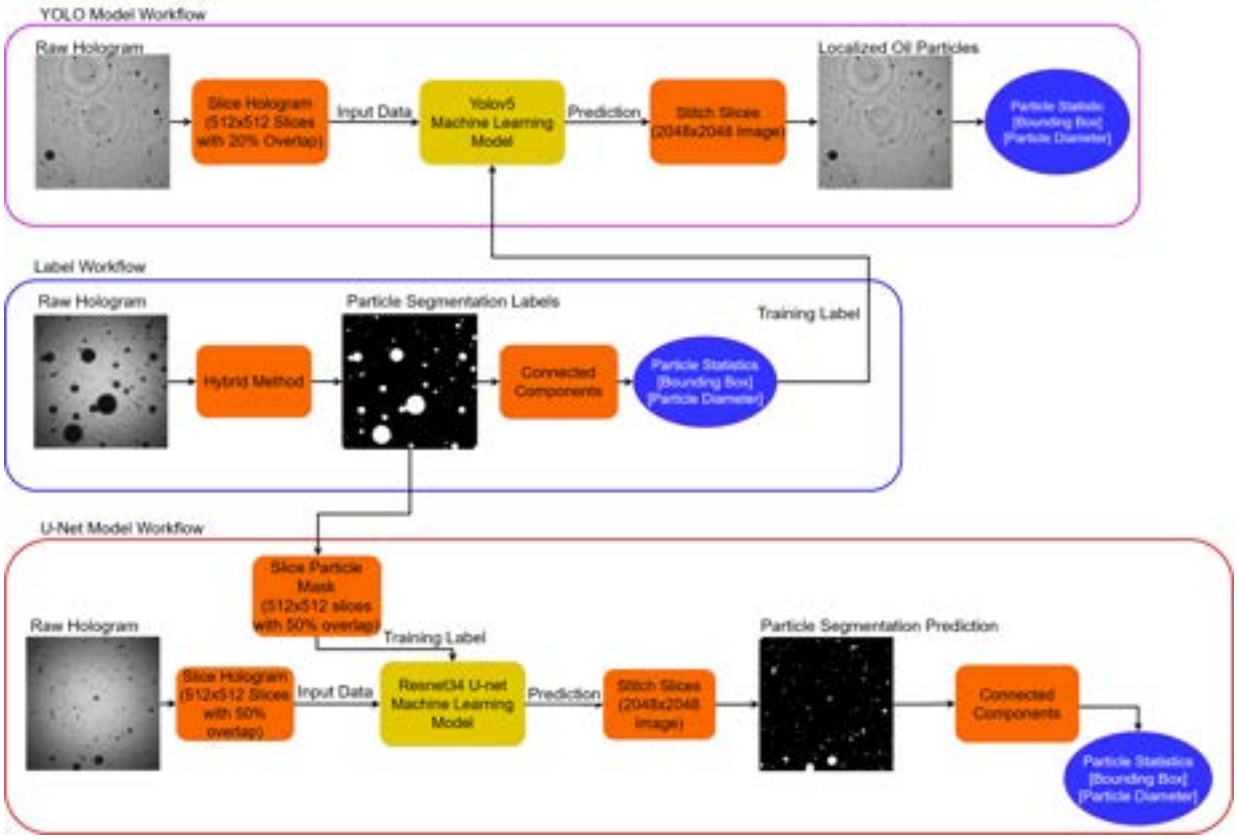
For the YOLO model, we used the python package slicing aided hyper inference (sahi) to create these cropped images [26]. We cropped the images into 512x512 sections with a 20% overlap. We overlapped the images to give the model a greater opportunity of predicting particles located between two cropped images in case one image contained more of the diffraction pattern than the other. Figure 3.5 outlines the preprocessing and postprocessing in our label generation and training workflow.



(a) Label particle mask (b) U-Net predicted particle mask

Figure 3.4: Prediction fidelity lost when downsampling the images.

Figure 3.5: Data Processing Workflow



Chapter 4

Results

Our initial goal was to test how effectively a machine learning model can learn the relationship between a raw hologram and the particles' shape and size. We generated 256x256 synthetic holograms with various concentrations of perfectly spherical oil droplets. Each hologram ranged from 0-25 particles with sizes ranging from 0.5-1.5 µm. We used a Mie scattering theory model to generate the holograms. We designed a convolutional neural network with 4 convolutional layers and 3 dense layers. Our label was the sum of the oil volume in each image. This is the total summation of the volume of each oil particle in a hologram. We chose oil volume as a metric to combine both size and particle quantity into a single value. Both the number of oil particles and the size of each particle had to be correctly predicted to generate the correct volume of oil. Our model had a root mean square error of 0.0344 on a test set. Figure 4.1 shows the results of the experiment. We demonstrated that the model was able to learn a relationship between particle shape and size based solely on raw synthetic holograms. We were unable to train the model successfully with the real-world dataset. The resulting model erroneously predicted the same volume for each hologram. We attributed this to the larger size and noise of real-world holograms.

Ultimately, we trained the U-Net and YOLO models on a GPU-accelerated server. The server employed a 24 GB RTX-3090 GPU, 128 GB of RAM, and an AMD 48-core threadripper CPU. The U-Net trained for 64 epochs. After 64 epochs, the loss on the validation set stopped decreasing. This took 22 hours. The YOLO model trained for 124 epochs. After 124 epochs, the loss on the validation set stopped decreasing. This took 48 hours.

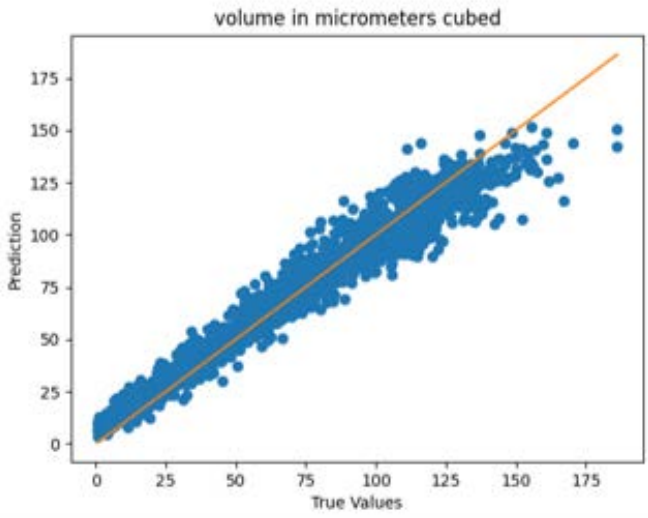


Figure 4.1: Results of the convolutional neural network experiment.

The output of the U-Net prediction is the particle mask. We applied connected components to extract bounding boxes from this particle mask. We used the bounding boxes to compare the U-Net with the YOLO model.

We split our dataset into 3 parts. We used 60% of the images for training (training set), 10% for validating the models during training (validation set), and 30% for testing the model performance after training (test set). We kept the model from the epoch with the best performance on the validation set. The model's performance metrics reported in this chapter are on the test set.

The metrics we used to assess the models are based on the Common Objects in Context (COCO) dataset evaluation metrics [27, 28]. These metrics are one of the most common metrics for object detection models. We used average precision (AP) and average recall (AR). Precision measures how often a prediction is a correct prediction. Recall measures how often a model made a detection when an object was present. Both of these metrics use intersection over union (IOU) to measure if a bounding box is a correct prediction or not. IOU above a determined threshold is considered a correct prediction. Precision and recall range between 0 and 1. Average precision is calculated as the averaged precision at

each recall r:

$$IoU = \frac{Area\,of\,Overlap}{Area\,of\,Union}$$

True Positive (TP)

False Positive (FP)

False Negative (FN)

True Negative (TN)

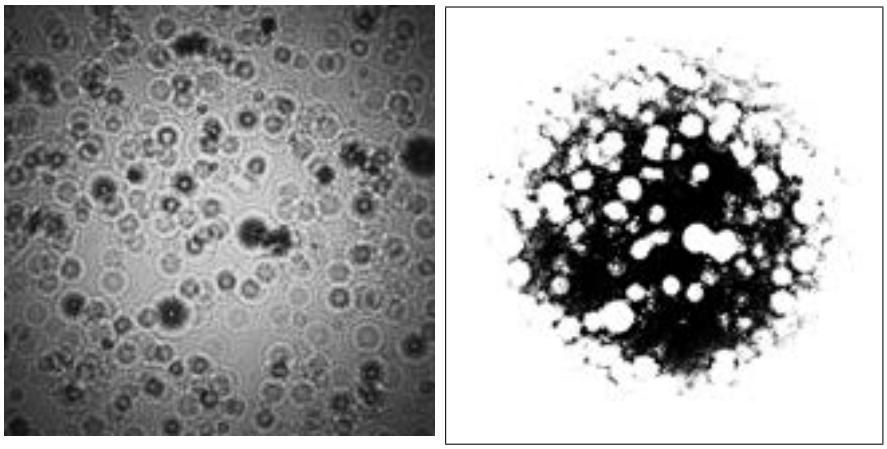
$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$
$$AP = \frac{1}{11} \sum_{r\epsilon\{0,0.1...,1\}} precision(r)$$

Average recall is double the area under the graph if recall is plotted with respect to IOU threshold, o:

$$AR = 2 \int_{0.5}^{1} recall(o)\, do$$

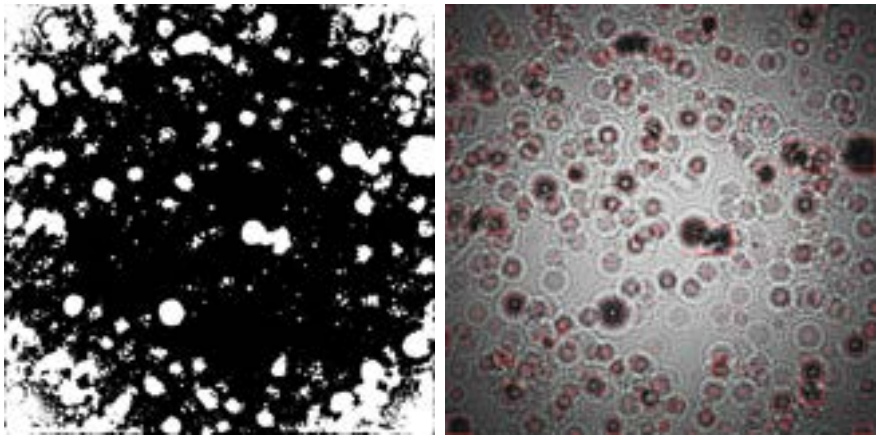
There is an inherent drawback to using hybrid method derived labels. Figure 4.2 shows an occasion in which the hybrid method failed. This occurred because the hologram is oversaturated with oil particles. This created a lot of background noise and made the diffraction patterns difficult to process. The models' predictions may even outperform the hybrid label. However, because we are using the hybrid method as labels, the models' evaluation will still be negatively impacted as they do not coincide with the hybrid method. We manually checked and removed samples such as these from the dataset. We used a large dataset to mitigate these errors. However, these errors will always occur to some degree in the dataset. Figure 4.5 shows an occasion where the machine learning models detected particles that the hybrid method did not detect.

The performance of both models is displayed in table 4.1. The YOLO network outperformed the U-Net in every metric. Comparing each model at AP@IoU={.75} and



(a) Raw hologram

(b) Hybrid method label



(c) U-Net mask prediction

(d) YOLO prediction

Figure 4.2: Failed hybrid method label and the models' performance on the hologram

AP@IoU=$\{.5\}$, there is a significant drop in AP. This shows that much of the loss in AP is due to inaccurate bounding box placements. While YOLO may be fairly accurate at detecting true positives, the model struggles to accurately measure the size of particles to a high degree. AR represents the model's ability to accurately count the number of particles.

Table 4.2 illustrates the execution time difference between the different methods. Because machine learning models are optimized for GPU processing, we recorded both with and without GPU acceleration. Both networks show drastic improvement in processing time. The YOLO model with GPU acceleration proved to be the fastest model. YOLO

Table 4.1: Model performance on a test set. Recall is averaged from the calculations when IoU threshold is set between 0.50 to 0.95. AP@IoU=0.75 and AP@IoU=0.5 is average precision at threshold 0.75 and 0.50 respectively.

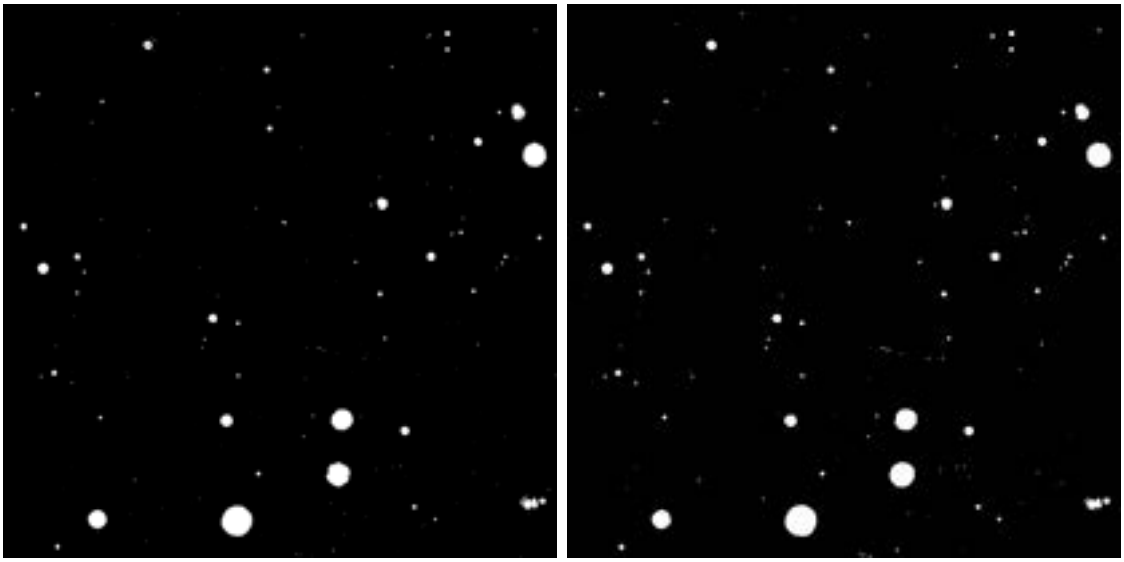
	AP@IoU={0.75}	AP@IoU={0.50}	AR@IoU={0.5:0.9}
Resnet34 U-Net	0.049	0.299	0.283
YOLOv5	0.418	0.717	0.516

Table 4.2: Time performance compared to the hybrid method

	Hologram processing time (s)	Time saved % from hybrid method
Hybrid method	165	0
Resnet34 U-Net (GPU)	3	98.2
Resnet34 U-Net (CPU)	5	97.0
YOLOv5 (GPU)	0.652	99.6
YOLOv5 (CPU)	370	-124

model without GPU acceleration was much slower than the hybrid method, with over a 100% increase in execution time.

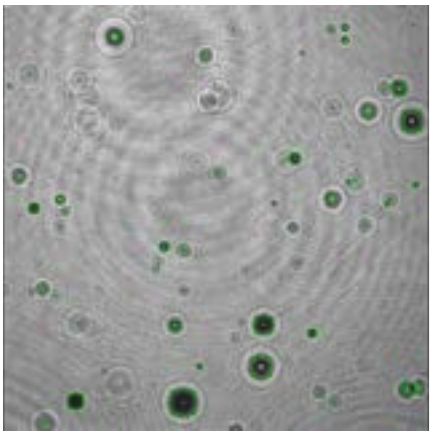
Figure 4.4 shows an example image from the test set. We overlayed the connected components bounding boxes onto the raw image for the U-Net prediction.



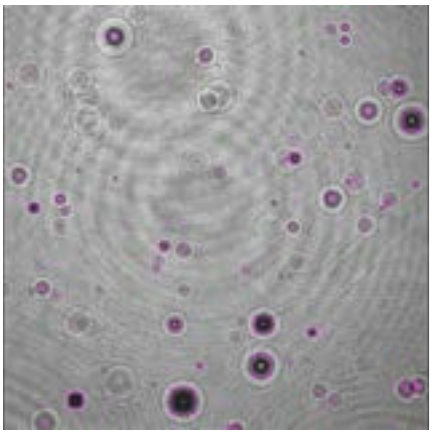
(a) Hybrid method generated particle mask label

(b) U-Net predicted particle mask

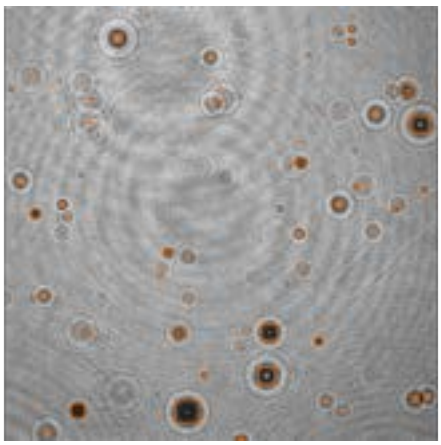
Figure 4.3: U-Net predicted particle mask comparison



(a) Hybrid Method

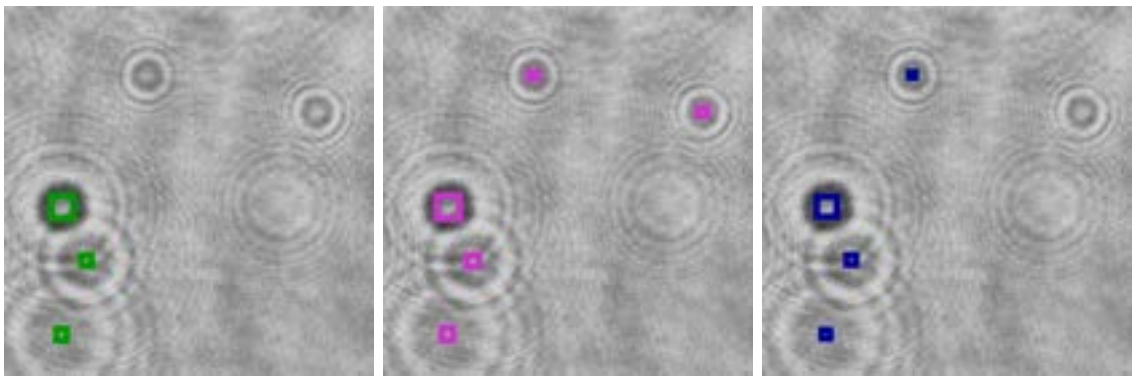


(b) U-Net



(c) YOLO

Figure 4.4: Example image from the test set.



(a) Hybrid Method (b) U-Net (c) YOLO

Figure 4.5: U-Net and YOLO detected particles that the hybrid method label did not detect

Chapter 5

Conclusion

Holographic sensors can record particles in a volume space without any need to adjust a focal plane. The hologram images preserve the particles' size and shape. With some processing, this data can be analyzed. We demonstrated a viable strategy to quickly process these images with the aid of machine learning. We tested two different approaches: instance segmentation with U-Net, and object localization with YOLO. The YOLO model outperformed the U-Net in every way. It had better AP and AR while also executing faster than the U-Net. The runtimes of both models are a significant improvement over the hybrid method. These methods show the potential for the holographic sensor to become a quick response sensor to assess oil spills in the ocean.

5.1 Future Work

Kukhtarev *et. al* [29], demonstrated that the diffraction patterns record the reflection coefficients. We can use the reflection coefficients to classify particles in a hologram image. Future work can extend the YOLO approach to classify other particles such as sand and air bubbles to avoid misclassifying these particles as oil particles.

When a particle occluded another particle, both models struggled to correctly separate the two particles. The hybrid method had a similar issue that was addressed in [17]. The updated algorithm used a depth map for each pixel belonging to a particle. The algorithm differentiated occluded particles based on the depth map. Both models can benefit from a similar technique. This depth information can be another input to the models. The models will use this information to separate partially occluded particles.

Combining the methods may produce better results. For example, we can use the machine learning model on the hologram image. We can use the hybrid method locally on the bounding boxes. The hybrid method will execute faster because it will reconstruct a smaller portion of the image. We will be able to use the precision of the hybrid method with a quicker execution time.

The labels were not 100% accurate because they were hybrid method derived. This created some uncertainty both during training and during validation that we mitigated by using a large dataset. Figure 4.2 suggests that a flatfield correction on the raw hologram images would improve the performance of the hybrid method and the machine learning models. This is due to the light intensity difference between the center and the corner of the image. One method would require collecting a few hologram images, averaging the holograms, then subtracting the holograms with this averaged image. It would also be beneficial to decrease the pixel variance found in a single hologram. We can do this by sampling a section in the center and in the corner of multiple holograms and dividing the holograms by this difference.

References

[1] P. J. Brandvik, Øistein Johansen, F. Leirvik, U. Farooq, and P. S. Daling, "Droplet breakup in subsurface oil releases – part 1: Experimental study of droplet breakup and effectiveness of dispersant injection," *Marine Pollution Bulletin*, vol. 73, pp. 319–326, 2013.

[2] L. DiPinto, H. Forth, J. Holmes, A. Kukulya, R. Conmy, and O. Garcia, "Three-dimensional mapping of dissolved hydrocarbons and oil droplets using a remus-600 autonomous underwater vehicle," Bureau of Safety and Environmental Enforcement (BSEE), Tech. Rep., 1 2020.

[3] S. Barkley, T. G. Dimiduk, J. Fung, D. M. Kaz, V. N. Manoharan, R. McGorty, R. W. Perry, and A. Wang, "Holographic microscopy with python and holopy," 2018.

[4] M. Adams, T. M. Kreis, and W. P. Jüptner, "Particle size and position measurement with digital holography," in *Optical Inspection and Micromeasurements II*. Society of Photo-Optical Instrumentation Engineers, 1997, vol. 3098, pp. 234–240.

[5] B. Schneider, J. Dambre, and P. Bienstman, "Fast particle characterization using digital holography and neural networks," *Applied Optics*, vol. 55, pp. 133–139, 2016.

[6] Y. Rivenson, Y. Wu, and A. Ozcan, "Deep learning in holography and coherent imaging," *Light: Science & Applications*, vol. 8, p. 85, 2019.

[7] Y. Pu and H. Meng, "An advanced off-axis holographic particle image velocimetry (hpiv) system," *Experiments in Fluids*, vol. 29, no. 2, pp. 184–197, Aug 2000.

[8] U. Schnars and W. Jüptner, "Direct recording of holograms by a ccd target and numerical reconstruction," *Applied Optics*, vol. 33, no. 2, pp. 179–181, Jan 1994.

[9] S. Murata and N. Yasuda, "Potential of digital holography in particle measurement," *Optics & Laser Technology*, vol. 32, no. 7-8, pp. 567–574, 2000.

[10] J. P. Fugal, T. J. Schulz, and R. A. Shaw, "Practical methods for automated reconstruction and characterization of particles in digital in-line holograms," *Measurement Science and Technology*, vol. 20, no. 7, p. 075501, may 2009.

[11] F. Soulez, L. Denis, C. Fournier, Éric Thiébaut, and C. Goepfert, "Inverse-problem approach for particle digital holography: accurate location based on local optimization," *Journal of the Optical Society of America A*, vol. 24, no. 4, pp. 1164–1171, Apr 2007.

[12] B. J. Thompson, "Holographic particle sizing techniques," *Journal of Physics E: Scientific Instruments*, vol. 7, no. 10, p. 781, oct 1974.

[13] Y.-S. Choi and S.-J. Lee, "Three-dimensional volumetric measurement of red blood cell motion using digital holographic microscopy," *Applied Optics*, vol. 48, pp. 2983–2990, 6 2009.

[14] L. Denis, C. Fournier, T. Fournel, C. Ducottet, and D. Jeulin, "Direct extraction of the mean particle size from a digital hologram," *Applied Optics*, vol. 45, no. 5, pp. 944–952, Feb 2006.

[15] J. Gao, D. R. Guildenbecher, P. L. Reu, and J. Chen, "Uncertainty characterization of particle depth measurement using digital in-line holography and the hybrid method," *Optics Express*, vol. 21, pp. 26 432–26 449, 11 2013.

[16] D. R. Guildenbecher, J. Gao, P. L. Reu, and J. Chen, "Digital holography simulations and experiments to quantify the accuracy of 3d particle location and 2d sizing using a proposed hybrid method," *Applied Optics*, vol. 52, pp. 3790–3801, 6 2013.

[17] J. Gao, D. R. Guildenbecher, L. Engvall, P. L. Reu, and J. Chen, "Refinement of particle detection by the hybrid method in digital in-line holography," *Applied Optics*, vol. 53, p. G130–G138, 9 2014.

[18] T. Shimobaba, T. Takahashi, Y. Yamamoto, Y. Endo, A. Shiraki, T. Nishitsuji, N. Hoshikawa, T. Kakue, and T. Ito, "Digital holographic particle volume reconstruction using a deep neural network," *Applied Optics*, vol. 58, pp. 1900–1906, 3 2019.

[19] S. Shao, K. Mallery, and J. Hong, "Machine learning holography for measuring 3d particle distribution," *Chemical Engineering Science*, vol. 225, p. 115830, 2020.

[20] C. Valadares, D. Brault, L. Denis, and C. Fournier, "Spherical object segmentation in digital holographic microscopy by deep-learning," in *Unconventional Optical Imaging II*, vol. 11351. Society of Photo-Optical Instrumentation Engineers, 2020, pp. 207–216.

[21] K. Mallery, D. Canelon, J. Hong, and N. Papanikolopoulos, "Design and experiments with a robot-driven underwater holographic microscope for low-cost in situ particle measurements," *Journal of Intelligent & Robotic Systems*, vol. 102, p. 32, 2021.

[22] Y. Zhang, Y. Zhu, and E. Y. Lam, "Holographic 3d particle reconstruction using a one-stage network," *Applied Optics*, vol. 61, no. 5, pp. B111–B120, Feb 2022.

[23] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. Jorge Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, M. J. Cardoso, T. Arbel, G. Carneiro, T. Syeda-Mahmood, J. M. R. Tavares, M. Moradi, A. Bradley, H. Greenspan, J. P. Papa, A. Madabhushi, J. C. Nascimento, J. S. Cardoso, V. Belagiannis, and Z. Lu, Eds., vol. 10553. Cham: Lecture Notes in Computer Science, 2017, pp. 240–248.

[24] P. Iakubovskii, "Segmentation models," https://github.com/qubvel/segmentation_models, 2019.

[25] Q. Shi, L. Cao, and Y. Wu, "High-speed three-dimensional digital holographic reconstruction of particles based on a yolo network," G. Volpe, J. B. Pereira, D. Brunner, and A. Ozcan, Eds., vol. 11804. Society of Photo-Optical Instrumentation Engineers, 2021, p. 1180417.

[26] F. C. Akyon, S. Onur Altinuc, and A. Temizel, "Slicing aided hyper inference and fine-tuning for small object detection," in *2022 IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 966–970.

[27] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision – European Conference on Computer Vision 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., vol. 8693. Cham: Springer International Publishing, 2014, pp. 740–755.

[28] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[29] N. Kukhtarev, T. Kukhtareva, and S. C. Gallegos, "Holographic interferometry of oil films and droplets in water with a single-beam mirror-type scheme," *Applied Optics*, vol. 50, pp. B53–B57, 2011.

Curriculum Vitae

Daniel Jonathan Cruz was awarded a Bachelor of Science in Electrical Engineering in 2019 at the University of Texas at El Paso. While earning his Electrical Engineering degree he was involved in several research groups including Biomedical Engineering Lab, Netlab, and System Ecology Lab. In 2018 he published an article titled "Bluetooth Enabled Smartphone Application For Wireless Photoplethysmography Monitoring Devices" to the Journal of the Mississippi Academy of Sciences. In 2019 he was awarded the Arctic Domain Awareness Center Fellowship where he worked on developing ecological sensors in Anchorage and Utqiagvik Alaska.

In 2020 Daniel Cruz enrolled in the Master of Science in Computer Science degree program. He continued his research with collaboration with the Vision and Learning Lab. In 2020 he was awarded the nation GEM consortium Fellowship which gave him the opportunity to pursue an internship at Oak Ridge National Laboratory where he applied machine learning techniques and novel data preprocessing techniques to identify cyber attacks on additive manufacturing devices. In 2021 he was awarded the NOAA Center for Earth System Science and Remote Sensing Technologies Fellowship. Through this fellowship, he served an internship at Woods Hole Oceanographic Institute's Scibotics Lab in collaboration with NOAA Office of Response and Restoration and University of New Hampshire Coastal Response Research Center to research machine learning applications in oil analysis.

main email: dcruzer231@gmail.com