Human Detection for Flood Rescue:
Application of YOLOv5 Algorithm and DeepSort Object Tracking

A Thesis Presented to
The Academic Faculty

by

Jessica Lewert

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Civil Engineering
School of Civil and Environmental Engineering

Georgia Institute of Technology
December 2021

Human Detection for Flood Rescue:
Application of YOLOv5 Algorithm and DeepSort Object Tracking

Approved by:

Dr. John E. Taylor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Eric Marks
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Adjo Amekudzi-Kennedy
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. James Forrest Toelle
Director of Information Technology
Columbus Consolidated Government

Date Approved: December 8, 2021

ACKNOWLEDGEMENTS

First and foremost, I would like to acknowledge my advisor, Dr. John Taylor for making this work possible and guiding me through each stage of research and writing. I am thankful for each member of my thesis committee, Dr. Eric Marks, Dr. Adjo Amekudzi-Kennedy, and Dr. James Forrest Toelle for taking the time to review my work. Additionally, I would like to thank the City of Columbus officials for their participation and support in this research, specifically Deputy Chief Daniel Hord, and Jeremy Miles. Finally, I am grateful for my family's continuous support in all my endeavors, especially my parents who always believe in me no matter what I challenges I may take on.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AP	Average Precision
CNN	Convolutional Neural Network
DP_NMS	Dynamic Programming with Non-Maxima Suppression
EMS	Emergency Medical Services
F_1	Precision-recall score
fps	Frames per second
GPU	Graphics processing unit
JDPA	Joint Probabilistic Data Association
LP2D	Linear Programming on 2D Coordinates
mAP	Mean Average Precision
MDP	Markov Decision Process
MHT_DAM	Multiple Hypothesis Tracking
MOT	Multi Object Tracking
NOMT	Near Online Method Tracking
PFD	Personal Flotation Device
SAR	Search and Rescue
SMOT	Single-shot Multi Object Tracking
SORT	Simple Online and Realtime Tracking
SOT	Single Object Tracking
TBD	Tracking By Detection
TC_ODAL	One-shot Distributed Algorithm for Logistic
TDAM	Temporal Dynamic Appearance Modeling
UAV	Unmanned Aerial Vehicle
USGS	United States Geological Survey
YOLO	You Only Look Once

SUMMARY

The city of Columbus, GA is situated directly on the Chattahoochee River along the Alabama-Georgia border. It is the second-largest city in Georgia (following Atlanta), with a population of roughly 200,000 people[1]. Columbus's proximity to the Chattahoochee provides residents and tourists with an abundance of outdoor river activities such as whitewater rafting, fishing, and kayaking. Frequent flooding, however, poses a threat to pedestrians who may find themselves stranded on the river following the release of water from upstream dams. From 2017-2019, 11 people drowned in the Columbus-Phenix City area of the Chattahoochee River. Columbus Fire and EMS reported an average of 18 rescue calls a year from January 2019 to November 2021, totaling 54 rescue calls over about three years.

This thesis proposes a method of human detection using high-resolution surveillance cameras paired with state-of-the-art object detection methods to monitor sections of the Chattahoochee River that require frequent search and rescue efforts due to flooding. The areas of interest are located in the city of Columbus, Georgia. The goals of this study are to evaluate the feasibility of the YOLO (You Only Look Once) algorithm for human detection on the Chattahoochee River as well as to propose future improvements to the city's existing alert methods in the event of a flood when upstream dam water releases and strands pedestrians.

[1] https://en.wikipedia.org/wiki/Columbus,_Georgia Wikimedia Foundation. (2021, October 20). *Columbus, Georgia*. Wikipedia. Retrieved November 17, 2021, from https://en.wikipedia.org/wiki/Columbus,_Georgia.

Testing was completed by recording river users in the determined areas of interest. Footage was passed through the model and analyzed to determine if the algorithms could successfully identify and track humans across frames. Results indicated that the YOLOv5 and DeepSort algorithm are feasible methods for detecting and tracking humans on the at-risk sections of the Chattahoochee River. The findings of this study as well as the review of relevant research may be used to further improve the safety conditions for river users in Columbus, Georgia.

CHAPTER 1: BACKGROUND

Existing Surveillance

Columbus currently has one functioning live webcam facing south along the Chattahoochee River, as seen in Figure 1. The live stream footage is available on the USGS (United States Geological Survey) website. The low resolution and placement of this camera make it unsuitable for human detection during flooding. It is placed too high above the ground to clearly distinguish pedestrians and fails to capture imagery of the areas where many rescues take place (downstream of two dams, the Lake Oliver Dam and North Highlands Dam). This thesis proposes the installation of multiple cameras along the Chattahoochee to better view the sections of the river where many have been stranded or drowned during flooding. The rock islands are where many river users find themselves stranded when the water levels begin to rise and pathways back to shore become submerged.

Figure 1: Live webcam stream of Chattahoochee River in Columbus

Accessed via www.usgs.gov

USGS also maintains a real-time stream gauge located near the live webcam. A description of the unit can be found in Table 1, and the location of the gauge can be seen in Figure 2. Available data includes air temperature/pressure, wind speed/direction, precipitation, relative humidity, discharge, and gauge height. Measurements are collected on the hour and results can be found on the USGS WaterData website.

Table 1: Description of 14th St Chattahoochee River Gauge

Description of Unit	Hydrologic Unit 03130003
Latitude	32°28’22”
Longitude	84°59’48”
Location	Muscogee County, GA

Datum of gauge	190.0 ft above NAVD88

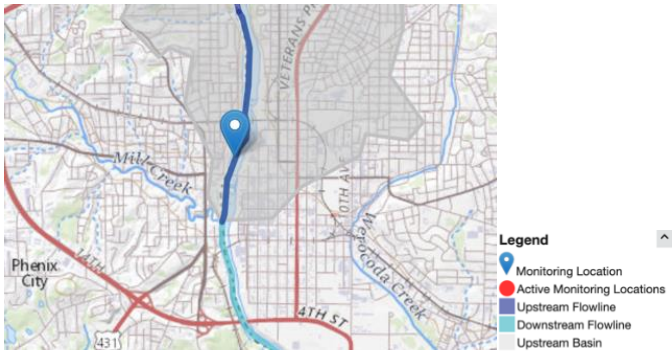


Figure 2: Map of Stream Gauge located at 14th St on the Chattahoochee River

Accessed via www.waterdata.usgs.gov

Further discussion of using real-time stream data for flood prediction and advanced alert systems can be found in the *Future Considerations* section of this thesis.

A Smart City

Recent efforts to promote safety for Columbus residents have been made in the form of "smart city" technology implementation. Some proposed advancements include free public Wi-Fi, license plate readers, public sensors to track population movement, and other Internet of Things (IoT) devices. The bustling Uptown district located at the city's center serves as the model region for a digital twin simulation, whereby "smart" technologies can be implemented at a smaller scale to then be implemented for the entire community. The end goal is to use collected data to improve quality of life and reduce emergency response time[2]. The addition of human detection methods at

[2] https://smartcities.gatech.edu/columbus *Columbus Consolidated Govt*. Smart Cities and Inclusive Innovation. (n.d.). Retrieved November 17, 2021, from https://smartcities.gatech.edu/columbus.

the Chattahoochee location ties into the city's desire to improve public safety and emergency response times through the use of smart technology.

A recent study in the Uptown area of Columbus demonstrated the efficacy of the YOLO algorithm when detecting pedestrians at the intersection of 10^{th} Street and Broadway. YOLO was able to detect and classify humans walking across the street to determine pedestrian exposure to heat, air pollution, and other environmental factors (Mavrokapnidis et al., 2021). Specific benefits of the YOLO algorithm will be further discussed in later sections.

River Rescue Efforts

From 2017-2019, 11 people drowned in the Columbus-Phenix City area of the Chattahoochee River. Columbus Fire and EMS reported an average of 18 rescue calls a year from January 2019 to November 2021, totaling 54 rescue calls over a period of about 3 years. It can be noted that 2020 saw a lower call volume likely due to the COVID-19 pandemic and the resulting reduction in number of river users. The nature of the calls ranges from drowning and recovery to victims stranded on the islands as water levels rise. Roughly 40% of calls reported since January 2019 occurred between 17:00 and 06:00. Analysis of previous rescue efforts suggests that it would be beneficial to employ the use of thermal imagery cameras to detect those in need of rescue after sundown. Human surveillance using thermal imagery and object detection will be discussed in later sections of this thesis.

The majority of search and rescue efforts occur following flooding caused by two dams along the Chattahoochee: the North Highlands Dam and the Lake Oliver Dam. The dams are operated by Georgia Power and maintain a somewhat regular schedule of release that fluctuates based on energy demands as well as rainfall conditions. A horn is sounded when water is released from the dams. Local whitewater rafting organizations are generally aware of the dam's schedule; however, independent fishermen, kayakers, and other tourists may be less knowledgeable of the horn's meaning or how long it may take to evacuate. Often, people stand on the cluster of rock islands, unaware of the rising water levels that follow opening of the dam, and ultimately end up stranded with no path to the shore once the river starts to flow more heavily.

The dam controllers receive a schedule of water release for the following day the night prior. When the dam is scheduled to open, a horn sounds to give river users roughly 15 minutes to get off the Chattahoochee. There is no signage explaining the sound alerts. However, when the alarm sounds there is an audible explanation stating that the water level is rising. Columbus Fire and EMS estimates that this alarm can be heard within a radius of a few hundred yards, but the noise of the river combined with other ambient conditions may have an impact on pedestrians' ability to hear the alarm. Figure 3 depicts a rescue effort made by Columbus Fire and EMS to save a man stranded on the rocks in the summer of 2019. Further discussion of using the tentative schedule to warn river users of incoming rising water levels can be found in the *Future Considerations* section.

Figure 3: Rescue effort by Columbus Fire and EMS

Accessed via www.chattvoice.com

City officials attribute the large number of incidents to both ignorance and a lack of respect for river guidelines. Keller and Gutscher attribute the inadequacy of flood response to two main causes: lack of individual risk perception and self-efficacy (Keller et al., 2006). As a result of several rescue incidents, personal flotation devices have been deemed mandatory for people on the river between the North Highlands Dam and the Columbus Iron Works Convention Center, and multiple signs have been placed on both sides of the Chattahoochee to inform river users of the law.

Columbus Fire and EMS identified three main areas of the Chattahoochee to be monitored due to their high level of traffic and history of rescue calls. Figure 4 displays the areas of interest in relation to the dams as well as the current USGS camera.

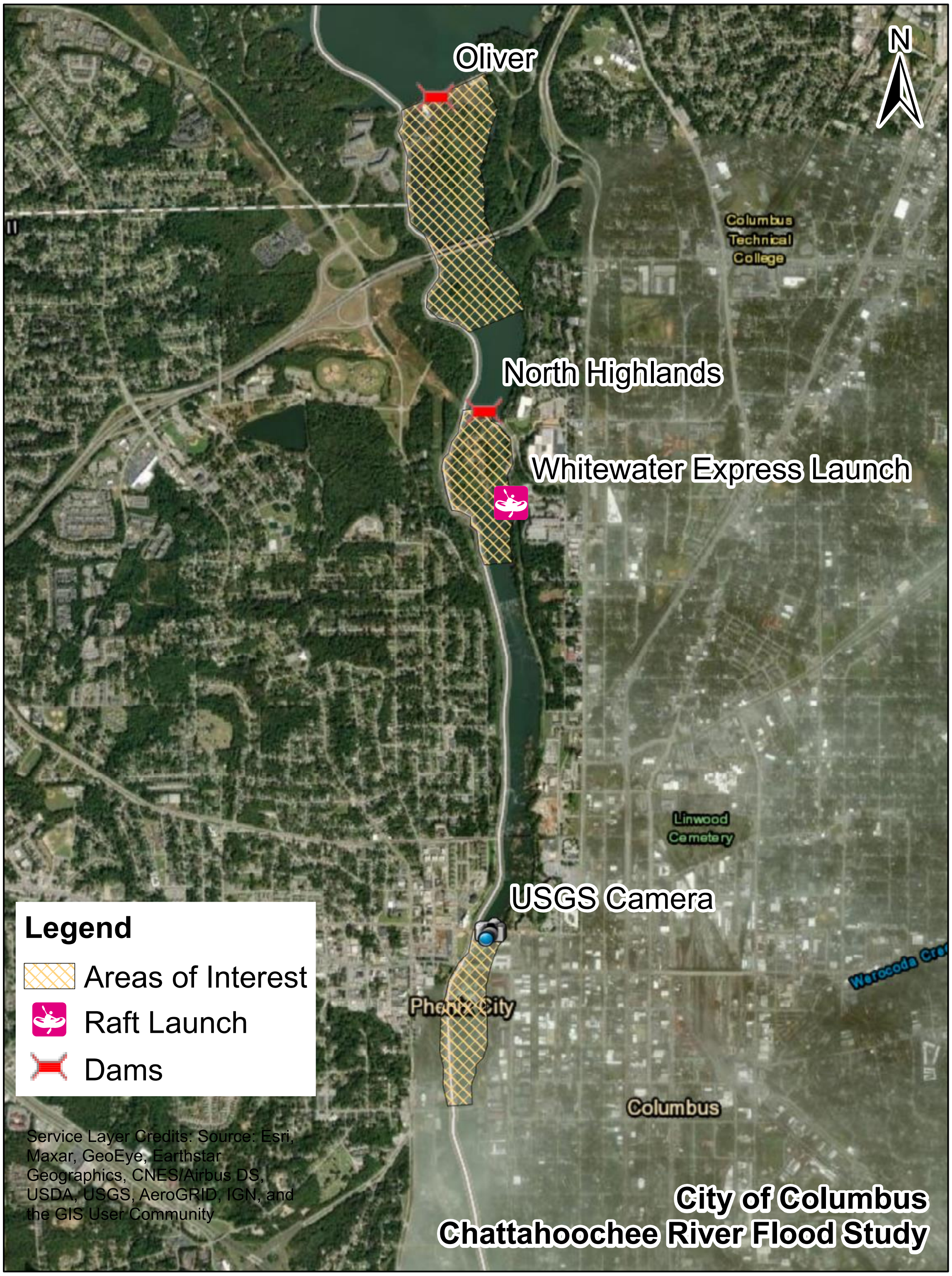


Figure 4: Areas of Interest for Human Surveillance in Columbus, GA

The objective of this thesis is to evaluate the effectiveness of the YOLOv5 algorithm for human detection paired with DeepSort object tracking. Chapter 2 discusses the decision to use YOLOv5 compared to other state-of-the-art object detection methods as well as briefly outline how the model works. Chapter 3 is an overview of the methodology and results followed by future considerations in Chapter 4. Finally, significance and conclusions can be found in Chapter 5.

CHAPTER 2: OBJECT DETECTION MODEL

Why YOLO?

The object detection algorithm chosen for the scope of this study is YOLO (You Only Look Once). This algorithm uses a single convolutional neural network (CNN) to quickly detect and classify objects. It was chosen for its accuracy in real-time as well as the public availability of its source code compared to other object detection methods. The structure allows for speedy object classification in real-time, at 45 frames per second, making this algorithm ideal for human detection on surveillance video in the event of an emergency or natural disaster (Redmon et al. 2016).

Compared to other state-of-the-art detection methods, such as Fast R-CNN, YOLO accounts for the entire image (rather than focusing on just one region of interest), leading to fewer background errors. Fast R-CNN is known as one of the best performing algorithms for object detection (Redmon et al. 2016). In a performance breakdown between Fast R-CNN and YOLO, Fast C-RNN was found to have nearly three times as many background errors as YOLO (see Figure 5).

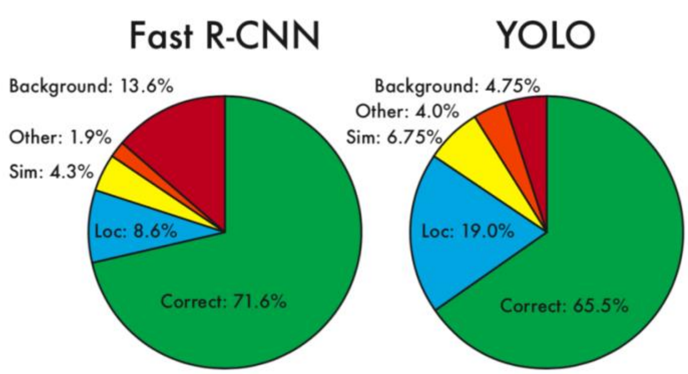


Figure 5: A comparative breakdown of YOLO and Fast R-CNN

Source: Redmon et al. 2016

Localization errors occur when an algorithm misaligns the predicted bounding box from the ground truth (the algorithm thinks the object is in a slightly different location than it is in reality). Background errors occur when the algorithm detects an object that does not exist. For the purposes of human detection in search and rescue, higher background accuracy outweighs the negative impacts of localization error.

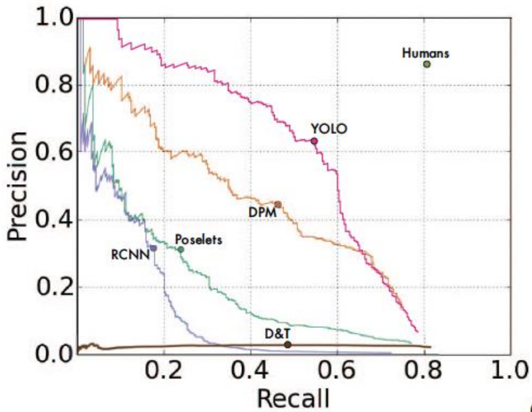
Fast R-CNN also falls short of the real-time time detection threshold (30 frames per second). Table 2 displays a comparison of state-of-the-art real-time object detection methods when tested on the well-known Pascal VOC 2007 dataset. YOLO achieved the highest mAP (mean Average Precision) value for real-time object detection. Fast R-CNN achieved higher mean average precision while sacrificing computing time and falling short of the real-time detection classification at 0.5 fps (frames per second) (Redmon et al. 2016). YOLO remains the best choice for object detection when considering real-time search and rescue needs.

Table 2: Real-time object detection comparison of leading methods

Real-Time Detectors	Train	mAP	FPS
100Hz DPM [30]	2007	16.0	100
30Hz DPM [30]	2007	26.1	30
Fast YOLO	2007+2012	52.7	**155**
YOLO	2007+2012	**63.4**	45
Less Than Real-Time			
Fastest DPM [37]	2007	30.4	15
R-CNN Minus R [20]	2007	53.5	6
Fast R-CNN [14]	2007+2012	70.0	0.5
Faster R-CNN VGG-16[27]	2007+2012	73.2	7
Faster R-CNN ZF [27]	2007+2012	62.1	18
YOLO VGG-16	2007+2012	66.4	21

Source: Redmon et al. 2016

Figure 6 displays another comparison of object detection algorithms on the VOC 2007 Picasso and People-Art Datasets as well as an evaluation of average precision (or AP) for leading object detection methods. Precision-recall is a performance metric of information retrieval/classification. The F_1 score is a weighted average of precision-recall. It can be seen that YOLO had the highest quality of performance of the tested datasets in both the AP and F_1 scores.



(a) Picasso Dataset precision-recall curves.

	VOC 2007	Picasso		People-Art
	AP	AP	Best F_1	AP
YOLO	**59.2**	**53.3**	**0.590**	**45**
R-CNN	54.2	10.4	0.226	26
DPM	43.2	37.8	0.458	32
Poselets [2]	36.5	17.8	0.271	
D&T [4]	-	1.9	0.051	

(b) Quantitative results on the VOC 2007, Picasso, and People-Art Datasets. The Picasso Dataset evaluates on both AP and best F_1 score.

Figure 6: Comparison of object detection methods on the Picasso and People-Art Datasets

Source: Redmon et al., 2016

Model scaling can improve accuracy and real-time inference depending on the type of device being used. Models are generally scaled by changing the depth and width of the convolutional neural network. Scaling is dependent on computing power or GPU (graphics processing unit) of the device being used. This technique may be considered for the Columbus project depending on the chosen implementation of the object detection system for search and rescue efforts (such as using YOLO on a mobile device or smaller computer for emergency efforts).

Authors Wang, Bochkovskiy and Liao demonstrate that a scaled version of YOLOv4 has achieved the highest average precision and speed amongst known state-of-the-art object detection methods for the well-known COCO dataset (a collection of annotated images with multi-object labeling categories used to represent the vast range of objects encountered in daily life), as seen in Figure 7 (Wang et al., 2011).

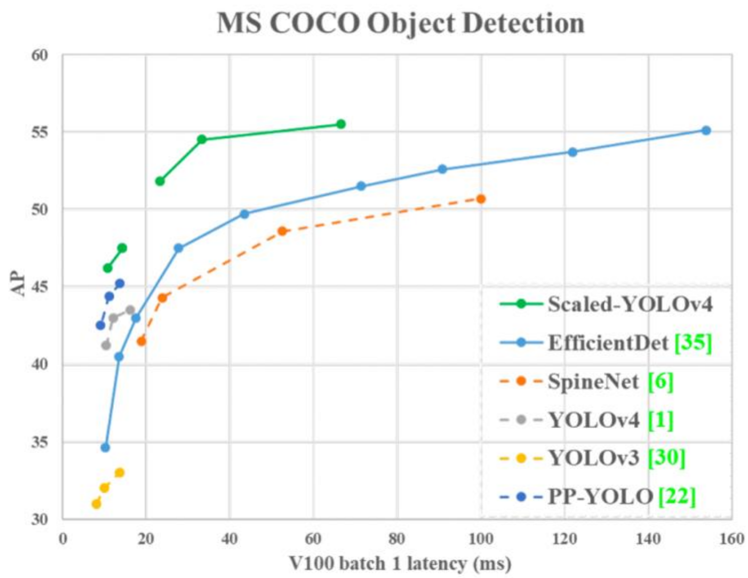


Figure 7: Comparison of object detection methods for the COCO dataset

Source: Wang et al., 2011

<u>YOLO Basics</u>

Research has indicated that YOLO remains a top choice algorithm for object detection due to its high speed and average precision, as well as its open-source availability. The functionality of the algorithm can be broken down into three basic steps, as seen in Figures 8 and 9.

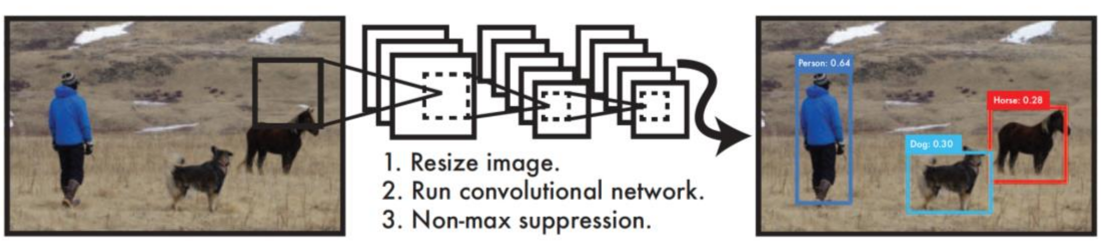


Figure 8: Object detection with YOLO

Source: Redmon et al., 2016

Figure 9 represents the three-step process used by YOLO to detect and classify objects. First, the image is divided into a grid of S X S dimension. Once the image is separated into a grid, each cell predicts the number of bounding boxes, represented by the variable *B*. Grid cells with objects fully enclosed are responsible for detecting the object within.

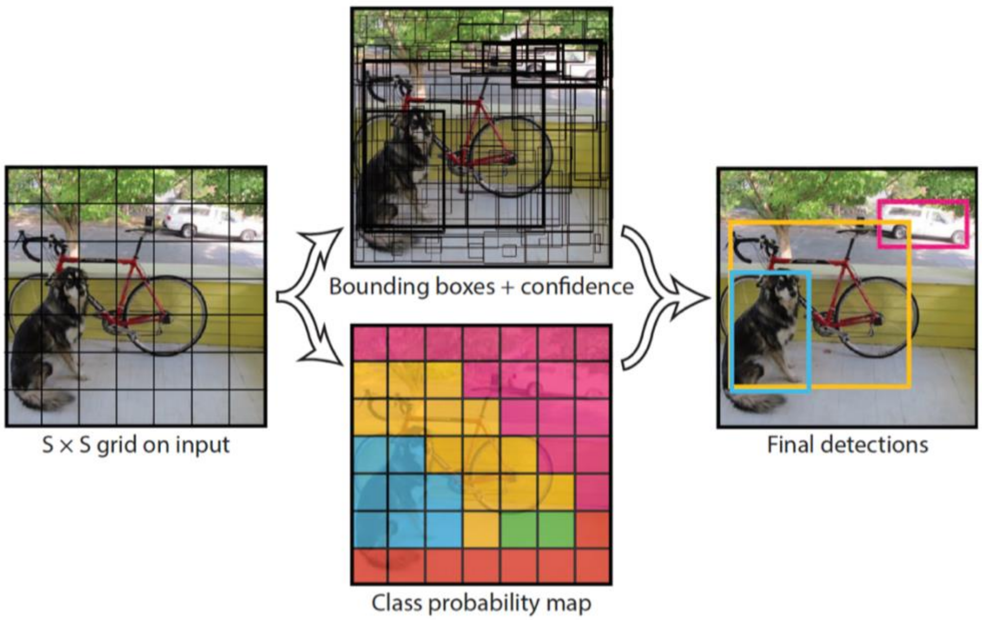


Figure 9: YOLO three step object detection process

Source: www.pyimagesearch.com

Confidence scores are predicted for each box, reflecting the level of confidence the model has for determining if the box indeed bounds an object, as well as the level of confidence in predicting what the object is. Confidence can be defined by the following expression:

$$\text{Pr(Object)} * IOU_{pred}^{truth}$$

Where *IOU* = intersection over union
truth = the number of ground truth boxes
pred = the number of predicted boxes

Figure 10 is a visual representation of the intersection over union computation. The ground-truth bounding box completely encompasses the stop sign, while the predicted bounding box determined by the algorithm is slightly displaced from the true object. The area of intersection between the two bounding boxes is the IOU.

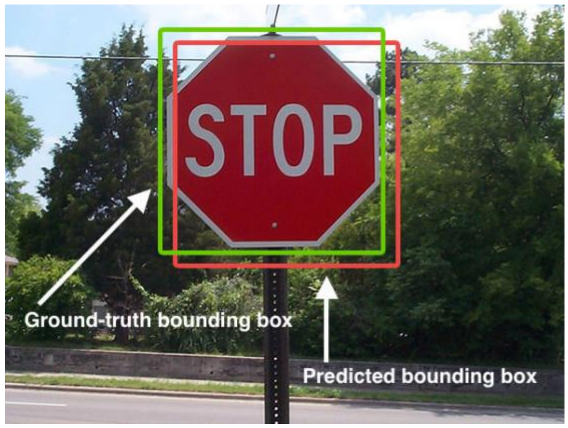


Figure 10: Intersection Over Union

Accessed via www.pyimagesearch.com

Grid cells without objects inside should have a confidence score of zero. Cells with objects should have a confidence score equal to the intersection over union between the algorithm's predicted bounding box and the ground truth bounding box. Within each bounding box there are five predictions: x, y, w, h, and the confidence. The x and y variables represent the center coordinates of the bounding box relative to the outside of the cell. The h and w variables are the respective height and width of the bounding box in comparison to the entire image, and the confidence is calculated as the intersection of union between the algorithm's predicted bounding box and the ground truth box.

Every one of the grid cells also predicts the conditional class probabilities, represented by the variable C. By multiplying the conditional class probabilities by the individual box confidence predictions, one can obtain confidence scores specific to each class for every bounding box. The model's predictions can be represented by the following tensor encoded into the algorithm and solved using regression (Redmon, 2016):

$$S \text{ x } S \text{ x } (B * 5 + \text{C})$$

This thesis seeks to explore how well the YOLO algorithm performs when used for human surveillance on the Chattahoochee River in Columbus. Specifically, it focuses on a recently updated version of the model, YOLOv5. The model is architecturally similar to previous releases but includes some changes such as it is written in Ultralytics PyTorch framework, which is known for intuitive use and extremely fast inferences[3]. This leads to faster training and batch inferences as can be seen in Figure 11.

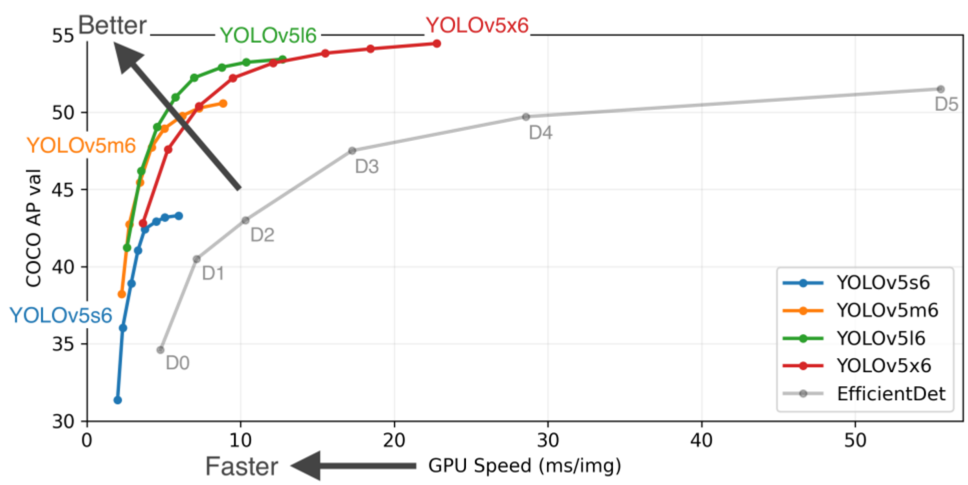


Figure 11: YOLOv5 performance on the COCO dataset

Source: www.towardsai.net

Additional improvements in the YOLOv5 update include augmented data training through a data loader, specifically scaling, color space adjustments and mosaic augmentation. The most notable

[3] https://blog.roboflow.com/how-to-train-yolov5-on-a-custom-dataset/ Solawetz, J. (2021, October 27). *How to train yolov5 on a custom dataset*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/how-to-train-yolov5-on-a-custom-dataset/.

of the three, mosaic augmentation, is specifically useful in addressing the model's small data problem in which smaller objects are undetected or improperly classified compared to larger objects[4]. The mosaic augmentation functions by simulating random crops, combining classes that are not frequently seen together, and varying the number of bounding boxes per image. Augmentations can improve model accuracy up to 10%[5] (Dwyer, 2020).

PyTorch and Convolutional Neural Networks

Previous versions of YOLO required the use of Darknet, an open-source neural network framework for real-ime object detection. YOLOv5, however, is written in PyTorch framework (Redmon, 2013). PyTorch is objectively more popular than Darknet based on referenced links on common data science/machine learning platforms[6]. The framework differences allow YOLOv5 to train faster and produce faster inference times than previous versions [7].

A Convolutional Neural Network (CNN) is a deep learning algorithm architecturally similar to the connectivity pattern of neurons within the human brain. "Convolution" is a mathematical operation on two functions to produce a third function, merging two sets of information. The algorithm can process an input image and assign learnable weights/biases to objects in the image, allowing for

---

[4] https://blog.roboflow.com/yolov5-improvements-and-evaluation/
Solawetz, J. (2021, September 21). *Yolov5 new version - improvements and evaluation*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/yolov5-improvements-and-evaluation/.

[5] https://blog.roboflow.com/advanced-augmentations/ Dwyer, B. (2020, October 5). *Advanced augmentations in Roboflow*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/advanced-augmentations/.

[6] https://www.saashub.com/compare-pytorch-vs-darknet *Pytorch vs darknet*. SaaSHub. (n.d.). Retrieved November 17, 2021, from https://www.saashub.com/compare-pytorch-vs-darknet.

[7] https://blog.roboflow.com/yolov4-versus-yolov5/
Nelson, J. (2021, March 4). *Responding to the controversy about yolov5*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/yolov4-versus-yolov5/.

object detection. Images are passed through a filter or kernel to create a feature map. At each location in the image, a convolution is performed (see Figure 12).

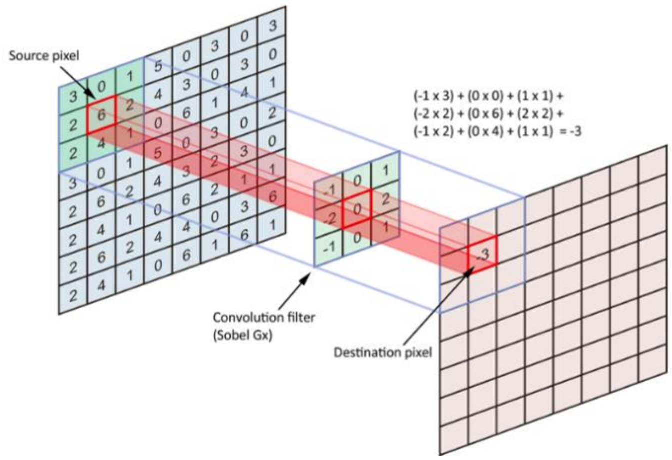


Figure 12: Filter application to source image in CNNs

Source: www.towardsdatascience.com

Convolutional Neural Networks differ from conventional Neural Networks in that they are organized into three dimensions (height, width, and depth). Unlike regular Neural Networks, the neurons within a CNN do not connect to all neurons in the next layer, only a small region of the layer, as can be seen in Figure 13[8].

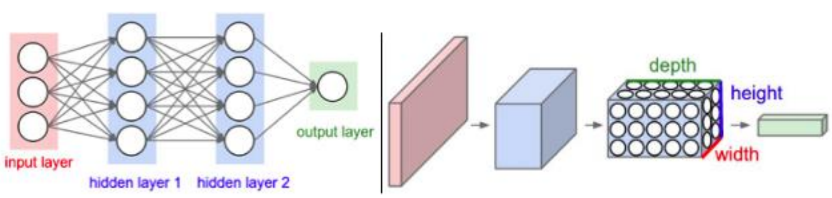


Figure 13: Regular Neural Network (left) compared to Convolutional Neural Networks (right)

Source: www.freecodecamp.org

[8] https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/
FreeCodeCamp.org. (2018, February 26). *An intuitive guide to Convolutional Neural Networks*. freeCodeCamp.org. Retrieved November 17, 2021, from https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/.

Convolutional Neural Networks can be represented by two major steps: feature extraction and feature classification. During feature extraction, the network will execute several convolutions by passing the source image through filter layers to detect features. During classification, the connected layers will be assigned a probability for the object detected to predict what the object is.

DeepSORT Object Tracking

The YOLO algorithm is a method of object detection that provides the means to recognize and classify what is seen in a given image. However, when considering moving subjects in surveillance video, it is necessary to provide a way to track the detected objects across all frames. Object tracking has seen multi-faceted use across several industries, ranging from professional sports to automated driving. The two types of object tracking are Single Object Tracking (SOT) and Multiple Object Tracking (MOT). This thesis focuses on MOT, in which the model will identify and keep track of every moving object—or in this case, person—on a frame-by-frame basis. The object tracking model used is called DeepSORT, which is often paired with the YOLO algorithm for the purposes of surveillance (Held et al., 2016).

DeepSort is a machine learning model that can be used to assign unique identifications to detected bounding boxes and follow their trajectory across frames. This model is a variation of Simple Online and Realtime Tracking (SORT) (Bewley et al., 2017). Like YOLO, the SORT model is renowned for its real-time speed and accuracy amongst competing state-of-the-art models. Figure 14 plots the multi-object tracking accuracy (MOTA) and speed of several benchmark model trackers against SORT, indicating that it is the leading option amongst competitors. Kapania et al.

(2020) presents the applications of DeepSort combined with YOLOv3 multi-object tracking for UAV (unmanned aerial vehicle) surveillance in New Delhi. DeepSort is an improvement on the SORT model that addresses limitations caused by object occlusion that cause the model to confuse the same object as two separate entities after passing behind an obstruction (Cochard, 2021). The model uses artificial intelligence to detect similarities between detected objects in previous frames to avoid identifying the same object as a new entity.

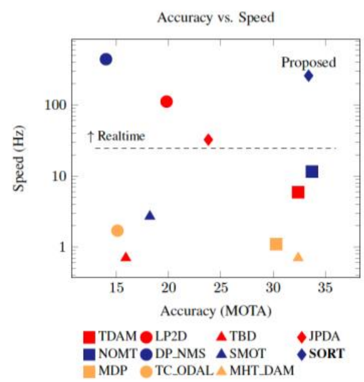


Figure 14: Performance of baseline model trackers[9]

Source: Bewley et al., 2017

The object tracking process can be broken down into three simple steps:

1) Determine the bounding boxes for detected objects using the YOLO algorithm
2) Assign identities (*ReID*) to each bounding box and apply the Kalman filter to project the trajectory of the objects between frames
3) Assign new identities to objects that were not previously detected

[9] See page 1 for explanation of abbreviations

Figure 15 displays a rudimentary flow diagram of the DeepSort object tracking process. In essence, the model will compare the bounding boxes between frames and determine if they are the same object based on the size and proximity of the box in the next frame. When detecting similarities between bounding boxes for subsequent frames, the Kalman filter is used to predict the future state of the detected object(s) based on current position (Punn et al., 2020). *ReID* is an identification model also used in DeepSort to determine the distance between feature vectors. Vector distances are calculated using previous frames on each track. Objects with the smallest distance between frames are assumed to be the same and assigned the same ID to distinguish them as the same entity across all frames.

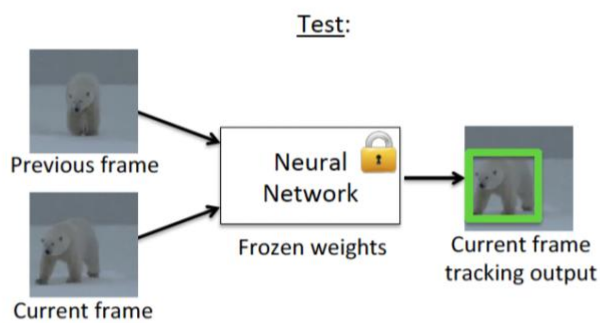


Figure 15: Flow diagram of DeepSort object tracking process

Source: Held et al., 2016

There are some challenges with the DeepSort model that are relevant when considering the placement of the camera for the purposes of this study. Occlusion of objects remains one of the top issues with tracking in surveillance[10]. Often, an object (in part) will pass behind another and the model will fail to detect it in the next frame. Figure 16 is an example of the occlusion issue,

---

[10] https://nanonets.com/blog/object-tracking-deepsort/?utm_source=medium&utm_medium=social&utm_campaign=3dhpe&utm_content=cv#multiple-object-tracking Maiya, S. R. (2020, April 24). *DeepSORT: Deep Learning to Track Custom Objects in a Video*. AI & Machine Learning Blog. Retrieved November 17, 2021, from https://nanonets.com/blog/object-tracking-deepsort/?utm_source=medium&utm_medium=social&utm_campaign=3dhpe&utm_content=cv#multiple-object-tracking.

whereby a detected person is about to pass behind a tree (left), then following the passing is not detected by the model (right). This shortcoming should be taken into consideration as far as camera placement for the Columbus project by choosing a location that minimizes potential for object occlusion.

Figure 16: Example of occlusion with the DeepSort tracking on the Columbus RiverWalk

Another issue to consider is variation in viewpoint should Columbus officials implement multiple cameras to observe one area. Object tracking may become difficult when the subject moves out of the frame into another viewport. This can be overcome by model training with networks such as the Siamese function (matches features of objects to maintain the same ID for an object in different views) but is not implemented for the scope of this study[11].

Training on a Custom Dataset

Training the YOLO algorithm to a custom dataset can result in higher accuracy and precision for detection. Due to the time constraints of this study, the video footage was tested on a pretrained

[11] https://nanonets.com/blog/object-tracking deepsort/?utm_source=medium&utm_medium=social&utm_campaign=3dhpe&utm_content=cv#multiple-object-tracking

model of YOLO. The algorithm was trained on the generic COCO dataset, which is generally recognized as the standard benchmark for human detection[12]. The COCO dataset contains 121,408 labeled images that are designed to represent generic objects seen in everyday life.

Figure 17 displays a breakdown of the 80 object classes for the COCO dataset. It can be seen that the object of interest for this study (people) is adequately represented. There are many existing publicly available datasets pertaining to human surveillance, such as the KTH human motion dataset, INRIA XMAS multi-view dataset, and the Weizmann dataset (Punn et al., 2020). Each contains a variety of basic categories of human action such as hand waving or running that may be pertinent to pedestrians in need of rescue.

Figure 17: Breakdown of COCO dataset object classes

Source: www.blog.roboflow.com

Future work may consider training the algorithm on images that may help YOLO better detect humans in a situation unique to the Chattahoochee River issue (such as training on swimmers, people halfway submerged in water, etc.) Additionally, if thermal cameras are installed, the training dataset should include thermal imagery such as the publicly available FLIR dataset.

---

[12] https://blog.roboflow.com/coco-dataset/ Solawetz, J. (2020, October 20). *An Introduction to the COCO Dataset*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/coco-dataset/.

Creating a set of training data requires use of annotation tools to create bounding boxes around objects in each image. The bounding boxes must then be labeled to represent each desired class. There are several publicly available annotation tools such as CVAT and Labellmg.

Assessing the Model

In training an algorithm for a given dataset, we can calculate the mean Average Precision or mAP by accounting for three types of losses: 1) classification loss, or errors in the prediction's accuracy, 2) localization loss, or errors between the predicted bounding box and the ground truth, and 3) confidence loss, or errors in the object's appearance in the bounding box. Mean Average Precision can be defined as the area under precision-recall curve, where precision and recall are defined by the following equations:[13]

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Models with high precision as recall increases can be considered well performing. The mAP takes the average precision over several intersection over union (IoU) thresholds (Liu et al., 2020). Setting an IoU threshold can influence the mAP value by restricting or expanding what the model reports as a detection (Ralašić, 2021). Figure 18 shows the differences in the precision-recall curve

[13] https://pro.arcgis.com/en/pro-app/latest/tool-reference/image-analyst/how-compute-accuracy-for-object-detection-works.htm *How the Compute Accuracy for Object Detection Tool Works*. ESRI. (n.d.). Retrieved November 17, 2021, from https://pro.arcgis.com/en/pro-app/latest/tool-reference/image-analyst/how-compute-accuracy-for-object-detection-works.htm.

as the threshold for IoU differs[14]. Figure 19 is an example of a mean average precision curve for training a model.

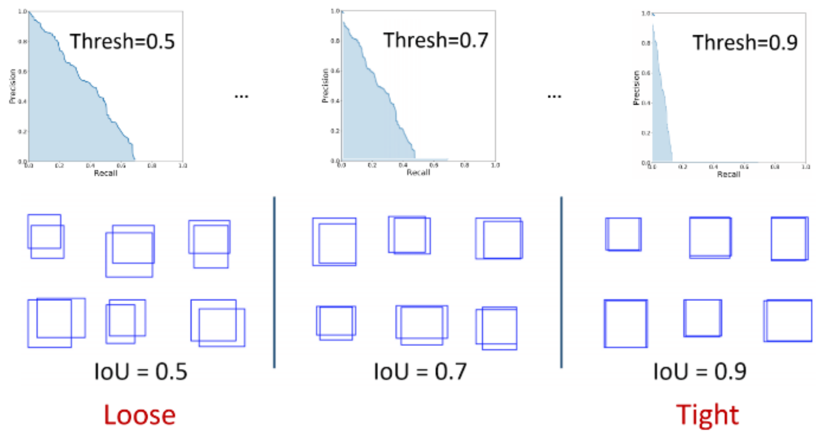


Figure 18: mAP variance as IoU value is restricted

Source: www.kharshit.github.io

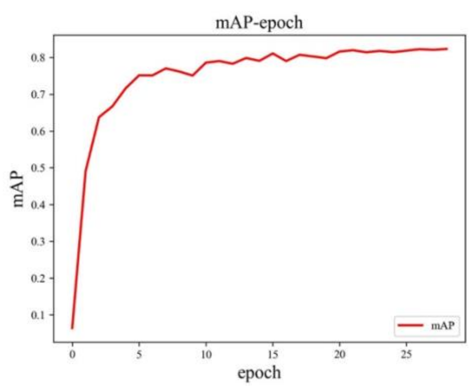


Figure 19: Example of mAP curve

Source: www.researchgate.net

Determining how many photos should be used to train the algorithm depends on evaluation of the mAP curve[15]. The dataset should be large enough that the algorithm maximizes its precision/recall,

[14] https://towardsdatascience.com/a-better-map-for-object-detection-32662767d424
Ralašić, I. (2021, October 13). *A better map for object detection*. Medium. Retrieved November 17, 2021, from https://towardsdatascience.com/a-better-map-for-object-detection-32662767d424.
[15] https://www.youtube.com/watch?v=MdF6x6ZmLAY&ab_channel=Roboflow
RoboFlow. (2020, June 14). *Yolov5 + Roboflow Custom Training Tutorial* . YouTube. Retrieved November 18, 2021, from https://www.youtube.com/watch?v=x0ThXHbtqCQ.

which can be seen at the point where the mAP levels out and is no longer increasing[16]. In Figure 19, this occurs around 20 epochs. Ways to improve model performance include adding more training images to the dataset or improving labeling (tightening bounding boxes).

[16] https://www.youtube.com/watch?v=MdF6x6ZmLAY&ab_channel=Roboflow
RoboFlow. (2020, June 14). *Yolov5 + Roboflow Custom Training Tutorial* . YouTube. Retrieved November 18, 2021, from https://www.youtube.com/watch?v=x0ThXHbtqCQ.

CHAPTER 3: MODEL TESTING

Methods

The purpose of testing for the scope of this study was to determine if the previously discussed model would be able to accurately identify humans in the areas of concern identified by Columbus Fire and EMS. The required sample imagery needed to contain footage along the river with at least one identifiable human. For the scope of this study, the visual footage (as opposed to infrared) was used to collect samples, requiring that video was taken during the daytime. Future work may include testing the model on nighttime imagery with an appropriate thermal camera.

The model was tested on two videos of the Chattahoochee River in areas of concern supplied by Jeremy Miles, Assistant Director of the Information Technology Department for the City of Columbus. The footage overlooks the rock conglomeration by the 14^{th} street pedestrian bridge, an area identified as "at-risk" by Columbus Fire and EMS. Each video was taken from a different vantage point (one on the sidewalk facing the rock conglomeration and one on the island where people can be found fishing or enjoying the view). The videos are roughly 30 seconds in length, taken in the afternoon on Friday, October 15, 2021.

Footage was tested in a Google Colab Notebook which provides free GPUs without need for computer configuration. The footage was imported into the notebook and tested using the pretrained YOLOv5 algorithm with DeepSORT tracking. After running the videos through the model, the resulting footage was analyzed to determine if the combination of YOLOv5 and DeepSORT was able to detect people on a frame-by-frame basis. Results of the footage testing can be seen in the next section. Recommendations for camera installation were based on the range and vantage point at which the model successfully identified people on the river and can be found in Chapter 4.

Results

Both tested videos showed successful identification of river users. Figures 20 and 21 show a comparison of frames between both videos before and after the algorithm was applied. The contribution of the YOLOv5 algorithm can be seen in the colored bounding boxes to identify each person in the frame, while the DeepSort model labeled each person with a unique number and tracked them throughout the duration of the video.

Figure 20: YOLOv5 applied to humans on rock conglomeration near 14th street

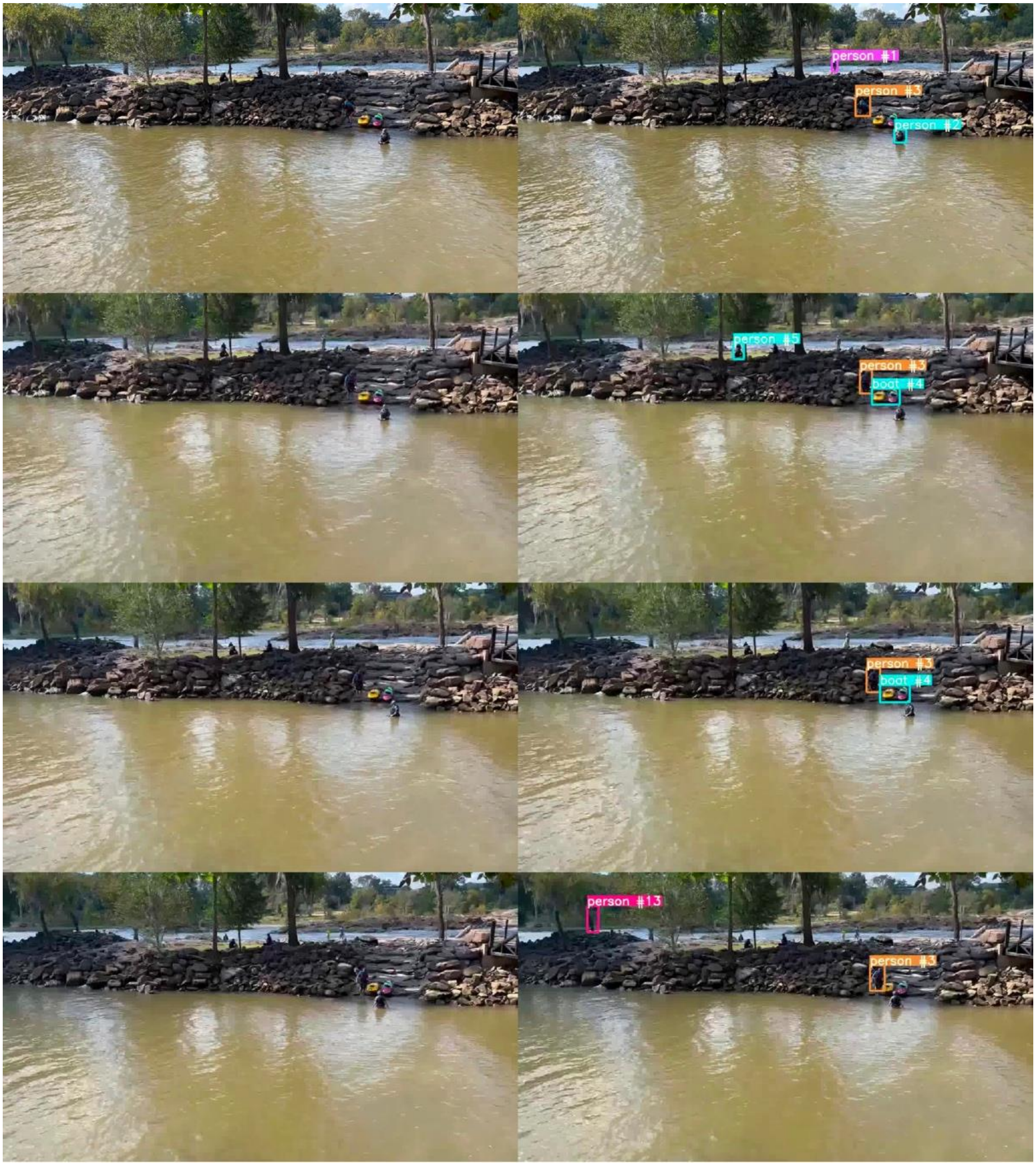


Figure 21: YOLOv5 applied to profile view of rock island near 14th street

It can be seen in Figure 21 that the YOLOv5 algorithm sometimes failed to detect partially obstructed humans (such as swimmers or people occluded by rocks) on a frame-by-frame basis. “Person #2” is identified in the first frame of Figure 21, but the algorithm fails to detect them in the next two frames. This is likely due to the fact that they are halfway submerged, and the COCO dataset does not contain sufficient imagery of swimmers to identify them with confidence. This may be remedied by further training the algorithm on a custom dataset, such as one that includes swimmers or humans in the water. It may also be noted that since the model was trained on the COCO dataset, which includes a variety of everyday objects, the algorithm was able to detect and label the boats in Figure 21.

For the purposes of this study, the COCO dataset allowed the YOLOv5 algorithm to successfully identify river users in most frames. Further training beyond the basic COCO dataset lies outside of the scope of this study, which is to simply evaluate the effectiveness of the YOLOv5 algorithm coupled with DeepSort object detection in detecting river users in the specified areas of interest. Overall, the tested model was able to identify nearly all humans in the footage although it lost track of some persons between frames. In the context of rescue applications, losing track of people for some frames is irrelevant if the person is identified over the majority of the footage (officials will not rely on a single frame to identify whether or not a person is present in an at-risk area, but rather they will view several hundred frames in real time). This was the case for this study; even if the bounding box was lost across a few frames, the algorithm was able to reidentify river users in a later frame. This object tracking issue can be addressed with enhanced model training in future studies.

CHAPTER 4: FUTURE CONSIDERATIONS

While the purpose of this study was to evaluate the effectiveness of the YOLOv5 and DeepSORT object detection/tracking model, a review of relevant research may be used in future work to improve the existing safety and surveillance system. This section will introduce alternative alert methods as well as briefly describe infrared imagery as it is relevant to this application.

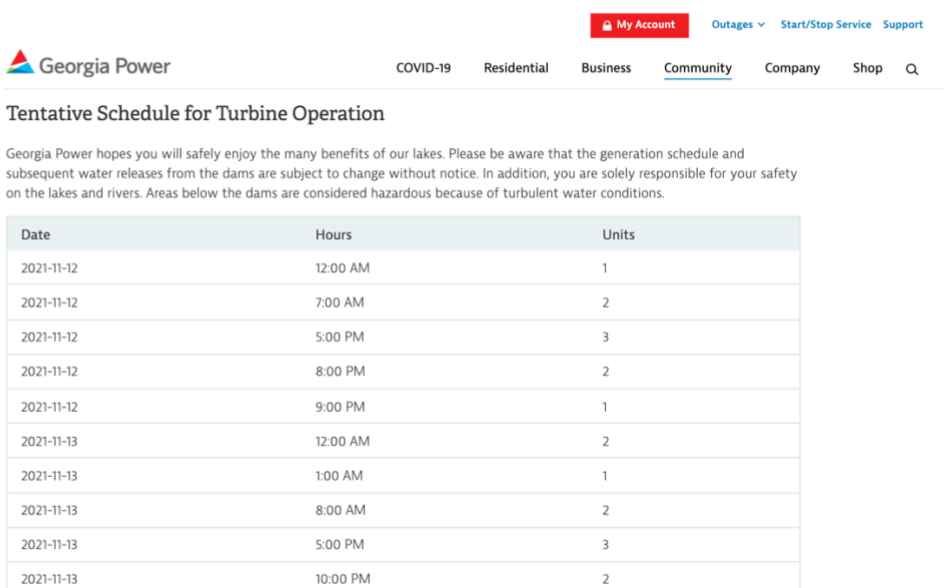
Alternative Alert Methods

Future consideration for research may include modification of Columbus's current alert system to increase public knowledge in the event of the flooding. Columbus relies on a siren to warn river users of oncoming flooding when the dam is scheduled to open. Despite this alarm, several rescue incidents and even fatalities continue to occur. Research indicates that sound-based alarms are more effective when coupled with further instruction. In a recent study evaluating the efficacy of various flood warning methods, Kuller, Schoenholzer, and Lienert address the benefits and limitations of sound-based flood alerts (Kuller et al., 2021). This method is often attractive for its low cost, effectiveness in capturing attention, and ease of installation/maintenance. However, sound-based alerts are often misunderstood or ignored without further explanation. Fakhruddin et al. proved that flood sirens are more effective when paired with additional information such as on-ground personnel or radio explaining the severity of the flood event and the need to evacuate (Fakhruddin et al., 2015) .

Georgia Power maintains an online tentative schedule of dam release for both the Lake Oliver and North Highlands Dams. Schedules are released for the current day as well as the following. A

Whitewater Express Rafting guide and Columbus resident explained that guides are expected to use the aforementioned schedule to determine when rafters may schedule a more or less "challenging" trip. However, it is unlikely that tourists or other river users are aware of the schedule's existence.

Figure 22 displays an example schedule of release accessible on the Georgia Power website. Table 3 contains a description of the flow amount in cubic feet per second (cfs) corresponding to the number of units in operation, as shown in the schedule in Figure 24. The Whitewater Expres raft guide described flows with two units operating to be turbulent and more challenging.



My Account Outages Start/Stop Service Support

Georgia Power COVID-19 Residential Business Community Company Shop

Tentative Schedule for Turbine Operation

Georgia Power hopes you will safely enjoy the many benefits of our lakes. Please be aware that the generation schedule and subsequent water releases from the dams are subject to change without notice. In addition, you are solely responsible for your safety on the lakes and rivers. Areas below the dams are considered hazardous because of turbulent water conditions.

Date	Hours	Units
2021-11-12	12:00 AM	1
2021-11-12	7:00 AM	2
2021-11-12	5:00 PM	3
2021-11-12	8:00 PM	2
2021-11-12	9:00 PM	1
2021-11-13	12:00 AM	2
2021-11-13	1:00 AM	1
2021-11-13	8:00 AM	2
2021-11-13	5:00 PM	3
2021-11-13	10:00 PM	2

Figure 22: North Highlands Dam Release Schedule

Accessed via www.georgiapower.com

Table 3: North Highlands Dam Unit Flow Descriptions

Units	Flow Amount (cfs)
1	800-1,000
2	5,000-8,000
3	9,000-12,000
4	12,000-13,300

Source: www.georgiapower.com

Future work may include investigating the benefits of installing electronic signage with the real-time schedules for each dam. Signs with current time, the daily schedule of dam release, and estimated water level rise could be placed near the areas of interest. Installation of updated signs would make river users aware of the expected release time and allow for anticipation of rising water levels, potentially preventing emergency rescues.

E. Tate and K. Cauwenberghs presented a method for web-based flood detection by compiling data from telemetered rain-gauge data combined with rainfall forecasts (Tate, Kauwenberghs, 2005). The data was then modeled to predict the areas most at risk and to what extent they may expect flooding. The simulation allowed local flood risk management to determine the best methods for river control operating strategies to mitigate potential damage caused by flooding. Additionally, the simulation publishes reports for public access to allow residents of flood risk zones to obtain warnings in the time leading up to a potential flood. Columbus Fire and EMS reported that weather data was not obtained for each rescue incident; however, it may be useful to analyze in future studies.

Another popular method of flood alerts is to broadcast via radio, SMS, TV or phone. Keoduangsine and Gardner-Stephen reviewed a variety of studies assessing the benefits of using SMS messaging to communicate the time of flood, predicted severity, and evacuation route (Keoduangsine, Gardner-Stephen, 2014). This method of alert is popular in developing countries due to the widespread use compared to the availability of the internet and real-time capability.

Infrared Imagery

Discussion with Deputy Chief Horton of Columbus Fire and EMS indicated that the team believes infrared imagery would be beneficial to rescue efforts. They are currently using thermal imagery in night calls to identify the bodies of those who have drowned in the Chattahoochee.

Thermal cameras are becoming increasingly popular for use in pedestrian detection and SAR (search and rescue) (Gade, Moesland, 2013). Sharma highlighted the benefits of using thermal imagery in UAVs to detect flood survivors (Sharma, 2016). Thermal cameras function by detecting thermal energy and use the information to create a visual image. These cameras are often preferred in situations where inclement weather may inhibit visibility. Thermal cameras do not require any additional light source, meaning they can function in conditions of low visibility, such as nighttime or in heavy fog.

Some setbacks to thermal imagery include sacrifice of detail due to the smaller range of detected temperatures as compared to the wider variety of visibly detectable color. Ambient temperatures may also inhibit thermal imagery should the subject and background temperatures be similar. This is referred to as “thermal crossover”. Davis and Sharma propose a method of fusing thermal and visible imagery to enhance background subtraction to combine the benefits of thermal and visible imagery into one frame, however for the scope of this thesis that method was not considered (Davis & Sharma, 2007).

Figure 23 shows the entire electromagnetic spectrum with the infrared range subdivided into its main categories. The FLIR ThermaCAM P10 covers the LWIR range between 7.5 and 13 mm. The sensor resolution is 320 x 240 pixels.

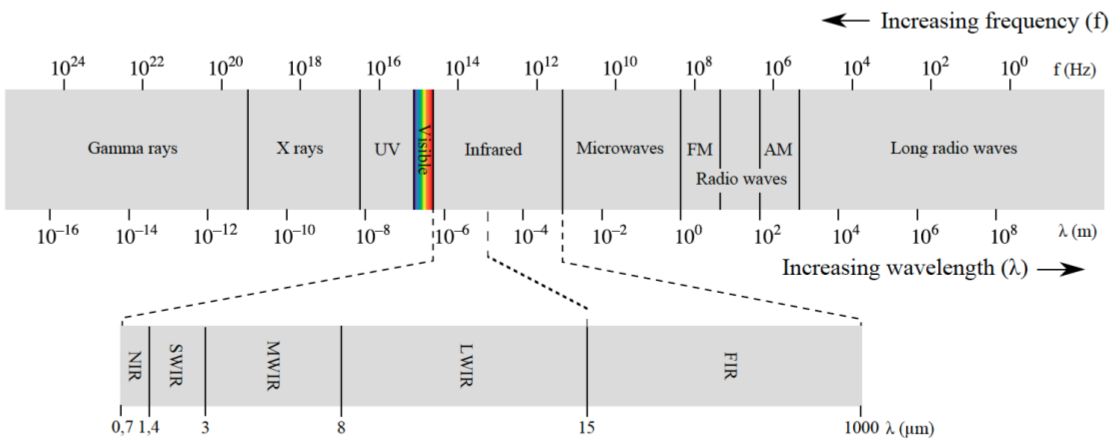


Figure 23: Electromagnetic spectrum with subdivided infrared range

Source: www.semanticsscholar.org

Researchers display the benefits of fitting the model with a telephoto lens for farther distances and in difficult weather conditions (Ivasic-Kos et al., 2019). Figure 24 shows a comparison of the image quality with and without the telephoto lens.

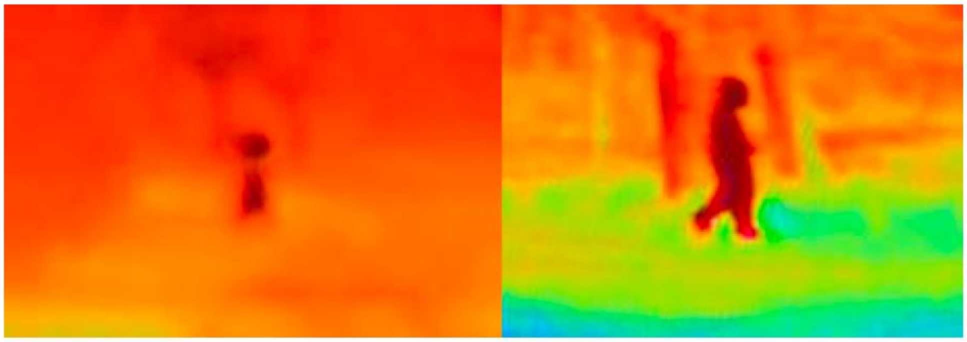


Figure 24: A human recorded with standard lens at a distance of 110 m (left) vs. telephoto lens recorded at 165 m (right)

Source: Ivasic-Kos, 2019

The YOLO algorithm was initially developed for object detection on RGB images (Kristo et al., 2020). However, as previously discussed, the algorithm can be trained to better suit a specific set of images with objects or conditions. If thermal cameras were implemented, it would be recommended that the algorithm be trained using an annotated dataset of thermal images. The FLIR dataset is a publicly available thermal image dataset with over 100,000 frames (28,151 of which are people)[17].

Identifying PFDs

Any person in the river section between the North Highlands Dam and the Columbus Iron Works Convention and Trade Center is required to wear a U.S. Coastguard approved personal flotation device, or PFD, under city ordinance. For the purposes of this study, the YOLOv5 algorithm was solely used to detect people. Extensive training on a dataset of personal flotation device images, or PFDs would allow emergency services to identify individuals who are not wearing the required

[17] https://www.flir.com/oem/adas/adas-dataset-form/

flotation gear and are therefore at a higher risk of drowning should the water levels rise. Should the number of persons identified by the algorithm not match the number of PFDs in a given frame, surveillance personnel could be alerted to apprehend the individual and issue a citation.

CHAPTER 5: CONCLUSION

The purpose of this study was to evaluate the efficacy of the YOLOv5 algorithm paired with DeepSort object tracking on footage of river users for flood safety surveillance. Testing indicated that the use of YOLOv5 with DeepSort tracking is a feasible model for detecting humans on the Chattahoochee River. Using a pretrained version of YOLOv5, the model successfully identified river users in specified areas of interest highlighted by Columbus Fire Rescue and EMS. Based on the results, cameras should be installed at a similar vantage point to that used in the model training dataset with a clear view of the area to be surveilled. There should be minimal to no obstruction of human view. Testing showed that using a high-definition camera, the algorithm was able to detect unobscured persons up to roughly 200 feet. Depending on the city's available budget, multiple cameras per area of interest may be used to gain a wider view of the at-risk areas. Installing cameras facing the rock conglomerations where many rescue efforts have occurred would allow for officials to be able to quickly identify at-risk individuals as well as efficiently locate those who are stranded on the rocks in a search and rescue event.

REFERENCES

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2017, July 7). *Simple online and realtime tracking*. arXiv.org. Retrieved November 17, 2021, from https://arxiv.org/abs/1602.00763.

Cochard, D. (2021, May 12). *DeepSort : A Machine Learning Model for Tracking People*. Medium. Retrieved November 17, 2021, from https://medium.com/axinc-ai/deepsort-a-machine-learning-model-for-tracking-people-1170743b5984.

Columbus Consolidated Govt. Smart Cities and Inclusive Innovation. (n.d.). Retrieved November 17, 2021, from https://smartcities.gatech.edu/columbus.

Davis, J. W., & Sharma, V. (2007, May 1). *Background-Subtraction Using Contour-Based Fusion of Thermal and Visible Imagery*. SemanticsScholar. Retrieved November 18, 2021, from https://www.semanticscholar.org/paper/Background-subtraction-using-contour-based-fusion-Davis-Sharma/bfedd15b5c82f97a7cc97df4754d1f255b58a12a.

Dwyer, B. (2020, October 5). *Advanced augmentations in Roboflow*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/advanced-augmentations/.

FLIR Thermal Dataset for Algorithm Training. Teledyne FLIR. (n.d.). Retrieved November 18, 2021, from https://www.flir.com/oem/adas/adas-dataset-form/.

freeCodeCamp.org. (2018, February 26). *An intuitive guide to Convolutional Neural Networks*. freeCodeCamp.org. Retrieved November 17, 2021, from https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/.

Gade, R., & Moeslund, T. (2013, January 1). *Thermal Cameras and aApplications:* . SemanticSCholar. Retrieved November 18, 2021, from https://www.semanticscholar.org/paper/Thermal-cameras-and-applications%3A-a-survey-Gade-Moeslund/74e2307a62404a86dbd454823f4a8c61f1df0f34.

Held, D., Thrun, S., & Savarese, S. (2016, August 16). *Learning to track at 100 fps with deep regression networks*. arXiv.org. Retrieved November 17, 2021, from https://arxiv.org/abs/1604.01802v2.

How the Compute Accuracy for Object Detection Tool Works. How the Compute Accuracy For Object Detection tool works-ArcGIS Pro | Documentation. (n.d.). Retrieved November 17, 2021, from https://pro.arcgis.com/en/pro-app/latest/tool-reference/image-analyst/how-compute-accuracy-for-object-detection-works.htm.

Informatics, M. I.-K. D. of, Ivašić-Kos, M., Informatics, D. of, Informatics, M. K. D. of, Krišto, M., Informatics, M. P. D. of, Pobar, M., & Metrics, O. M. V. A. (2019, April 1). *Human Detection in Thermal Imaging Using YOLO*. Human Detection in Thermal Imaging Using

YOLO | Proceedings of the 2019 5th International Conference on Computer and Technology Applications. Retrieved November 18, 2021, from https://dl.acm.org/doi/abs/10.1145/3323933.3324076.

Keoduangsine, S., & Gardner-Stephen, P. (2014, January). *A Review of Flood Warning Systems in Developed and Developing Countries*. ResearchGate. Retrieved November 18, 2021, from https://www.researchgate.net/publication/283585507_A_Review_of_Flood_Warning_Systems_in_Developed_and_Developing_Countries.

Kuller, M., Schoenholzer, K., & Lienert, J. (2021, July 21). *Creating effective flood warnings: A Framework from a critical review*. Journal of Hydrology. Retrieved November 18, 2021, from https://www.sciencedirect.com/science/article/pii/S0022169421007587.

Liu, Q., Jia, P., & Sun, Y. (n.d.). *Detection and Classification of Astronomical Targets with Deep Neural Networks in Wide Field Small Aperture Telescopes*. ResearchGate. Retrieved November 18, 2021, from https://www.researchgate.net/publication/339940039_Detection_and_Classification_of_Astronomical_Targets_with_Deep_Neural_Networks_in_Wide_Field_Small_Aperture_Telescopes.

M. Krišto, M. Ivasic-Kos and M. Pobar, June 18, 2020. "Thermal Object Detection in Difficult Weather Conditions Using YOLO," in *IEEE Access*, vol. 8, pp. 125459-125476, 2020, doi: 10.1109/ACCESS.2020.3007481.

Maiya, S. R. (2020, April 24). *DeepSORT: Deep Learning to Track Custom Objects in a Video*. AI & Machine Learning Blog. Retrieved November 17, 2021, from https://nanonets.com/blog/object-tracking-deepsort/?utm_source=medium&utm_medium=social&utm_campaign=3dhpe&utm_content=cv#multiple-object-tracking.

Mavrokapnidis, D., Mohammadi, N. and Taylor, J. (2021). “Community dynamics in smart city digital twins: A computer vision-based approach for monitoring and forecasting collective urban hazard exposure,” *In Proceedings of the HICSS, 54th Hawaii International Conference on System Science, Decision Analytics & Service Science*, Jan 2021 hdl.handle.net/10125/70832.

Nelson, J. (2021, March 4). *Responding to the controversy about yolov5*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/yolov4-versus-yolov5/.

Punn, N. S., Sonbhadra, S. K., & Agarwal, S. (2020, May 6). *Monitoring covid-19 social distancing with person detection and tracking via fine-tuned yolo V3 and Deepsort techniques*. arXiv.org. Retrieved November 17, 2021, from https://arxiv.org/abs/2005.01385v2.

Pytorch vs darknet. SaaSHub. (n.d.). Retrieved November 17, 2021, from https://www.saashub.com/compare-pytorch-vs-darknet.

Ralašić, I. (2021, October 13). *A better map for object detection*. Medium. Retrieved November 17, 2021, from https://towardsdatascience.com/a-better-map-for-object-detection-32662767d424.

Redmon, J. (n.d.). *Open source neural networks in C*. Darknet. Retrieved November 17, 2021, from https://pjreddie.com/darknet/.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016, May 9). *You only look once: Unified, real-time object detection*. arXiv.org. Retrieved November 17, 2021, from https://arxiv.org/abs/1506.02640.

RoboFlow. (2020, June 14). *Yolov5 + Roboflow Custom Training Tutorial* . YouTube. Retrieved November 18, 2021, from https://www.youtube.com/watch?v=x0ThXHbtqCQ.

Sharma, S. (2016, January 1). *Flood-Survivors Detection Using IR Imagery on an Autonomous Drone*. SemanticScholar. Retrieved November 18, 2021, from https://www.semanticscholar.org/paper/Flood-survivors-detection-using-IR-imagery-on-an-Sharma/048c3193942a9fa6aa416b669b9a3dc72167ab2b.

Shivani Kapania Bharati Vidyapeeth's College of Engineering, Kapania, S., Engineering, B. V. C. of, Dharmender Saini Bharati Vidyapeeth's College of Engineering New Delhi, Saini, D., Bharati Vidyapeeth's College of Engineering New Delhi, Sachin Goyal Bharati Vidyapeeth's College of Engineering New Delhi, Goyal, S., Narina Thakur Bharati Vidyapeeth's College of Engineering, Thakur, N., Rachna Jain Bharati Vidyapeeth's College of Engineering New Delhi, Jain, R., Preeti Nagrath Bharati Vidyapeeth's College of Engineering New Delhi, Nagrath, P., & Metrics, O. M. V. A. (2020, January 1). *Multi Object Tracking with UAVs Using DeepSort and YOLOv3 RetinaNet Detection Framework*. Multi Object Tracking with UAVs using Deep SORT and YOLOv3 RetinaNet Detection Framework | Proceedings of the 1st ACM Workshop on Autonomous and Intelligent Mobile Systems. Retrieved November 17, 2021, from https://dl.acm.org/doi/10.1145/3377283.3377284.

Solawetz, J. (2020, October 20). *An Introduction to the COCO Dataset*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/coco-dataset/.

Solawetz, J. (2021, October 27). *How to train yolov5 on a custom dataset*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/how-to-train-yolov5-on-a-custom-dataset/.

Solawetz, J. (2021, September 21). *Yolov5 new version - improvements and evaluation*. Roboflow Blog. Retrieved November 17, 2021, from https://blog.roboflow.com/yolov5-improvements-and-evaluation/.

Tate, E., & Cauwenberghs, K. (2005, September). *An innovative flood forecasting system for the Demer Basin: A case study*. ResearchGate. Retrieved November 18, 2021, from https://www.tandfonline.com/doi/abs/10.1080/15715124.2005.9635255.

Wikimedia Foundation. (2021, October 20). *Columbus, Georgia*. Wikipedia. Retrieved November 17, 2021, from https://en.wikipedia.org/wiki/Columbus,_Georgia.