MASTER THESIS

A visual analysis framework for dinghy sailing: Towards leveraging recorded training sessions

This thesis is submitted in partial fulfillment of the requirements for the degree of Master of Science

in

Computer Science

by

Gijs Martinus Wilhelmus REICHERT
born in Alkmaar, the Netherlands

Abstract

A visual analysis framework for dinghy sailing: Towards leveraging recorded training sessions

by Gijs Martinus Wilhelmus REICHERT

Nowadays video plays an important role in the coaching of athletes across many different sports. To make more use of the advantages videos can provide for coaching, the Dutch Sailing team is shifting from manually recording short videos towards continuously recording training sessions. This new recording approach provides opportunities and creates challenges at the same time. In this thesis we present a pipeline to address the problems with the stability of the recording and the first steps towards a Visual Analysis Framework, which leverages the available video data. New information is extracted from the video recordings by detecting and tracking the boat and sailors. Moreover, we semi-automatically highlight interesting intervals in time of a recorded training session. These are the first steps towards an extensive Visual Analysis Framework which has the potential to make the analysis of the videos easier and provide the coaches with tools to improve the analysis of the performance of the sailors.

Thesis Committee:

Prof.dr. A. Vilanova	TU Delft/ TU Eindhoven
Dr. R. Marroquim	TU Delft
Dr. J. van Gemert	TU Delft
G. van der Heijden	Annalisa B.V.
D. Broekens	Sailing Innovation Centre

Acknowledgements

First of all, I would like to thank my supervisors Anna Vilanova and Ricardo Marroquim. I am very grateful for their valuable feedback on my writing and their feedback during our weekly meetings. Next to this, I would like to thank Gert van der Heijden, Douwe Broekens, Marcos Pieras Sagardoy, Roy van Ophuizen and the coaches in Scheveningen for their help during my thesis. Moreover, I would like to thank all the people at the Nationaal Topsportcentrum Zeilen/Sailing Innovation Centre for making the days their such a nice experience. Next, I would also like to thank Jan van Gemert for completing the Thesis Committee.

Furthermore, I would like to thank all my friends, teammates and roommates in Delft and other places for their support and adventures together. All the skiing trips and nights out may not have been beneficial for the pace of my study but were always more than worth it. Moreover, I thank my cousin Jules and our oldtimer Bessie for the countless adventures and hours of fun. Similarly, I thank Jules, Maaike and Antoinette for helping keeping Bessie alive and for our adventures throughout Europe.

Last, but most certainly not least, I would like to thank my entire family. In particular, I would like to thank my parents Stan en José and my sisters Florine en Marjolein, for always supporting me and providing everything I could possibly need and more.

Contents

List of Figures

List of Abbreviations

FOV	Field of View
ITF	Interframe Transformation Fidelity
LoG	Laplacian of Gaussian
MSE	Mean-Squared-Error
NTCZ	Nationaal Topsportcentrum Zeilen
PPHT	Progressive Probabilistic Hough Transform
PSNR	Peak Signal-to-Noise Ratio
RAM	Random-Access Memory
RIB	Rigid-Inflatable Boat
SVM	Support-Vector Machine

Chapter 1

Introduction

Nowadays technology and data analytics are becoming increasingly intertwined with sports. The review by Barris and Button [1] reveals that for numerous sports vision-based analysis approaches exist, mostly focused on player movement and location. The data can help assess performance during training and in competitive settings. For example, players could receive video-based feedback which has been studied using ice-hockey players in the work of Nelson, Potrac, and Groom [2]. Video analysis in sport is also used for example to provide feedback [3], provide biomechanical analysis [4] and in sport psychology to motivate and build confidence [5]. It can even provide new insights that were not previously detected without the use of technology, such as for example field positions over time and fatigue in football. This trend holds for the complex sport of sailing as well, where more and more sensors are added to the boats to measure the performance. However, for most sailors in the Olympic dinghy class it holds that the use of sensors during training is not standard practice and not allowed during races. To record and review their performance the coaches and athletes make use of video to support their review and analysis.

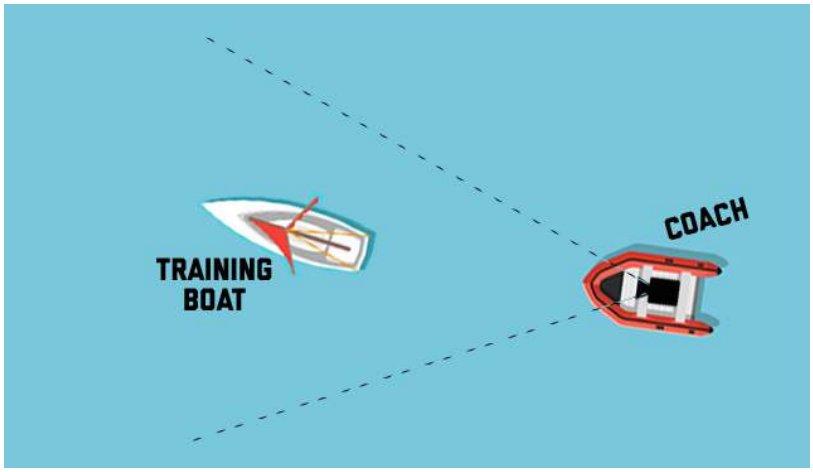


FIGURE 1.1: Schematic representation of coach following a boat during a training session.

When the athletes of a dinghy class boat go out to train the coach usually follows them around in a rigid-inflatable boat (RIB), a schematic representation can be found in Figure 1.1. Then, to gather valuable data the coach makes short video clips, usually not more than a minute per clip, of moments which highlight the goal of the training or provide videos to reflect on. Also, during starts of races the coaches tend to record videos to be able to assess the performance afterwards. These videos are shot using handheld cameras or smartphones, which despite some advantages, such

as only capturing the moments they think are interesting, also causes a number of problems. Problems such as, for example, the need to hold and operate them using one or two hands and the transition time from operating the RIB to being ready to record.

An example of a reasonable quality input video under acceptable weather conditions and sea state (the general condition of the water surface) can be seen in Figure 1.2, these videos were, however, not stabilized. Acceptable conditions will be discussed in more detail in Section 1.1.2.

Because most of the videos are shot using handheld devices, the quality regarding stability and observable details, such as rudder movement, is relatively low. Next to this, the coach has to use two hands to properly operate a RIB. This means that whenever the coach decides to capture a moment, one hand is operating the boat and the other is used to shoot videos. Sometimes important moments are not captured, due to the time it takes to switch between operating the boat and shooting a video of a sailor. Moreover, because the coach can not operate the RIB with two hands they tend to keep more distance between the sailing boat and the RIB to be safe. The increased distance between the camera and the sailing boat makes observing details from the videos, such as for example rudder movement, difficult. Although the video data could provide a wealth of information, the current approach leads to the video data not living up to its true potential.

FIGURE 1.2: Example footage taken from coach boat under good conditions.

The company annalisa[1] is making an effort to tackle some of these problems by mounting a camera to the coach boat. These cameras can then record the entire training session, which means that the coach could focus on operating the RIB while coaching. However, a training session usually lasts about 2-3 hours. Going through all of the footage after every training would be undesirable and too time-consuming for the coaches and sailors. Therefore, we propose a semi-automatic way to determine which parts are potentially worth reviewing and storing. To facilitate this we need a tool to be able to visually explore and analyse the footage. This research will aim to improve the stability of the video data for visual analysis purposes, detect interesting intervals in the footage and extract and/or cluster valuable data from the videos.

[1] https://annalisa-sailing.com/

1.1 Research Goal and Requirements

1.1.1 Research Goal

For the Dutch Olympic sailing teams some of the coaches already use video to support their analysis of training sessions and races. However, as mentioned previously, this system is far from ideal and the process varies from coach to coach. The goal of this research is to improve the visual analysis of videos used to train sailing athletes in the dinghy class. First, the stabilization of the footage taken from the coach boat that follows athletes will be improved. Stabilizing the footage should make it easier to analyze and observe details. Next, of the entire recorded session only the potentially useful parts for the coach and athletes will be highlighted and segmented. This means another goal is to determine what useful parts of a session are and develop a method to reliably detect and segment these from the video feed. Finally, we will provide a visual analysis strategy to facilitate the analysis of the video.

This research will provide the first steps towards what could be an extensive visual analysis framework for the coaches and athletes. The goal is to design the framework in such a way that only the video frame data is used and does not rely on external sensors. Next to this, another goal of this research is to require minimal manual user input to analyse relevant video sequences. This is because, according to some of the coaches, if it takes them more time than what they are used to now it is unlikely that they will ever use it. Therefore, in this thesis we will investigate how to semi-automatically process, analyze and segment recorded sessions while using only the video data.

1.1.2 Requirements and Assumptions

To limit the amount of variables for developing the methods in this thesis a few requirements and assumptions are in order. This thesis focuses on the situations where a Rigid-hull inflatable boat follows one 49er sailing boat. We assume to always be following one 49er boat under normal weather conditions. By normal weather we mean that there is no rain limiting the camera view and a sea state calm enough to record videos without extreme movements limiting the ability to keep the boat in view most of the time. Next to this, we assume that while following one boat no other boats will cross paths or be in the video next to the sailing boat of interest. Lastly, the recording camera needs to have a high enough resolution, with a minimum of 480p but preferably over 1280x720 pixels, and needs to be close enough to be able to see the persons and boat clearly in the video. This comes down to a maximum distance of around 30-40 meters, but preferably the RIB is following the sailing boat more closely.

1.2 Outline

Chapter 2 provides basic background information on sailing and the current and new situation. Chapter 3 gives an overview of relevant prior work in video analysis and sailing. Chapter 4 describes the proposed pipeline and used methods. Chapter 5 presents the evaluation of the proposed methods, which consists of experiments and evaluations with users. Chapter 6 concludes this thesis and discusses potential future work. This thesis is best viewed in color.

Chapter 2

Background

This thesis spans multiple topics and the purpose of this chapter is to provide the necessary basic knowledge and terminology used in this thesis. The background information mostly focuses on sailing.

2.1 Sailing

2.1.1 Boat Type

The methods and approaches in this thesis are focused on the 49er/49er FX (see Figure 2.1), a boat in the Olympic dinghy sailing class. Nevertheless, we assume that the methods should work for other dinghy class boats as well. This skiff type boat is operated by a two person crew, both equipped with their own trapeze. These trapezes are used to hang overboard and counteract the force of the wind by using their body-weight. The two person crew consists of the helm, the one who steers the boat, and the crew, who sits more towards the front of the boat and does most of the sail control.

FIGURE 2.1: The 49er schematic and real appearance. (Source: Wikimedia Commons and Watersportverbond)

Boat parts In Figure 2.2 several boat parts have been labeled. These relevant boat parts and their corresponding names will be referred to throughout this thesis. Some of these boat parts are important in understanding the manoeuvres, which will be discussed next.

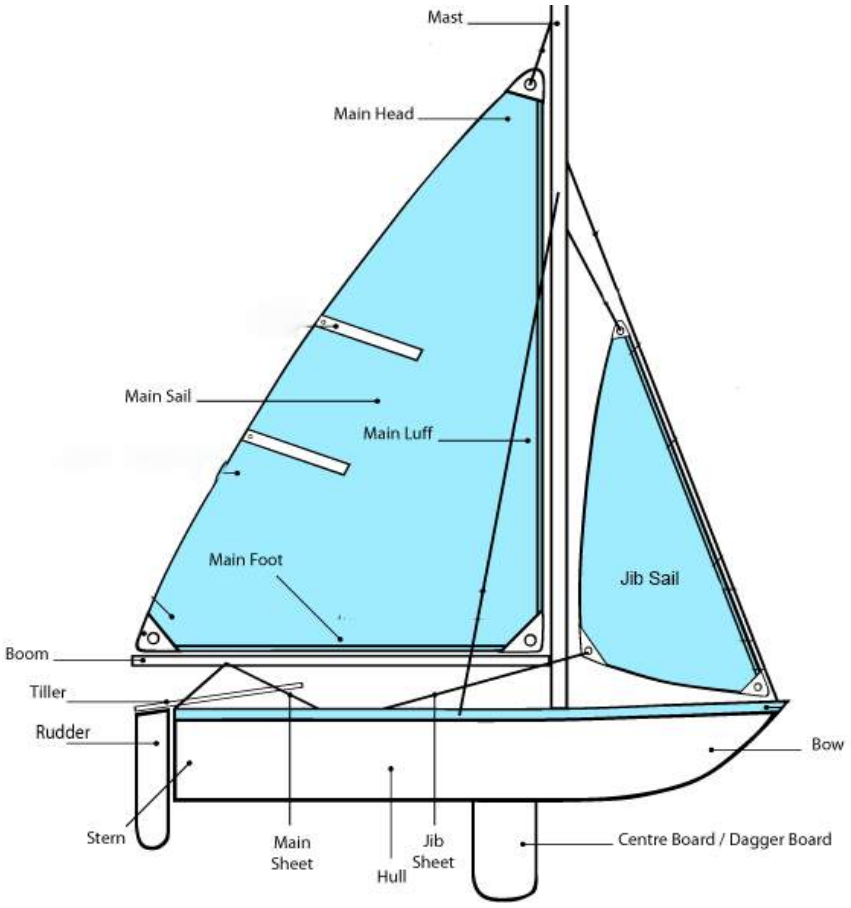


FIGURE 2.2: Relevant parts and names of a sailing dinghy. (Modified from Wikimedia Commons)

2.1.2 Manoeuvres

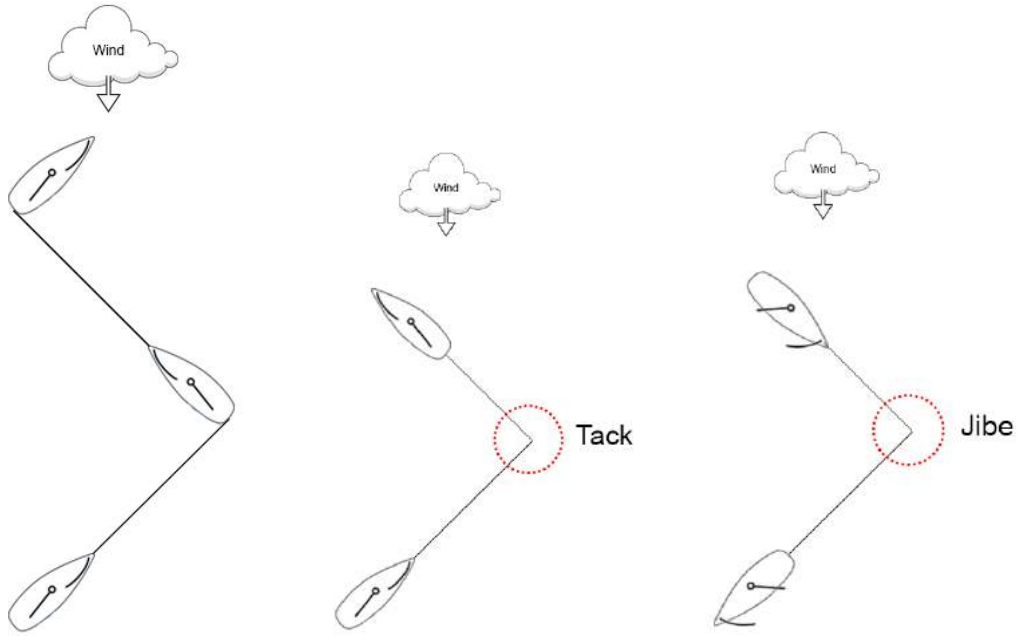
In sailing there are many different manoeuvres, but two stand out as most important in general and in the context of this thesis. These two manoeuvres are called Tacking and Jibing and will be explained in more detail below.

Tacking

You can not sail a boat directly into the direction the wind is coming from. This may, however, be the direction the boat needs to go. To overcome this limitation, the boat is sailed in a zig-zag pattern which is called beating. This is where the tacking manoeuvre is used as can be seen in Figure 2.3a. Tacking is when you steer the bow (see Figure 2.2) of the boat, using the rudder, in the direction the wind is coming from and continue turning until the wind is coming from the other side relative to the boat. A schematic representation of a Tacking manoeuvre can be seen in Figure 2.3b. During this manoeuvre the sail and sailors move to the other side, relative to the imaginary centerline from the bow to the stern of the boat.

Jibing

Jibing is the opposite of Tacking. With the stern (see Figure 2.2) of the boat facing the wind direction the boat is turned "through the wind". For a schematic representation of a jibing manoeuvre, see Figure 2.3c.



(A) Beating to windward. (B) Tack Manoeuvre. (C) Jibe Manoeuvre.

FIGURE 2.3: Beating to windward using tacks, and common manoeuvres tacking and jibing.

2.1.3 Hiking

Although hiking is not a manoeuvre, it is an important part of dinghy sailing. Hiking, also known as leaning out or sitting out, is the action of counteracting the wind force using your body weight and thereby reducing the heel angle of the boat. Hiking in the 49er boat is more "extreme", as the crew is further away from the hull with their body, which should make detecting the location of the person with respect to the boat easier. The detection of the location of the person with respect to the boat is used to extract manoeuvres from the videos, which will be discussed in Chapter 4. An example of hiking can be seen in Figure 2.4.

FIGURE 2.4: Crew of a 49er hiking, green outline highlighting the location of the sailors.

2.1.4 RIB Usage

As mentioned before, in Chapter 1, the coach usually follows a boat around during a training session in a RIB. This is done in order to stay close enough for the coach to be able to assess the performance and actions of the sailors. To operate the RIB you need two hands, one for the throttle handle and one for the steering wheel. The throttle handle (see Figure 2.5 on the side of the console, left of the steering wheel) can, however, be left in a position which frees up one hand to do something else, leaving the throttle handle in the same position. Although the coach spends quite some of the time following the boat around, the distance between the coach boat and the sailboat varies. The coach in the RIB could be at around 5-10 meters behind when taking a closer look at the sailors actions or be further behind to have a better overview. Moreover, sometimes the coach and the sailboat are stationary and side-by-side in the water in order for the coach to provide feedback during training.

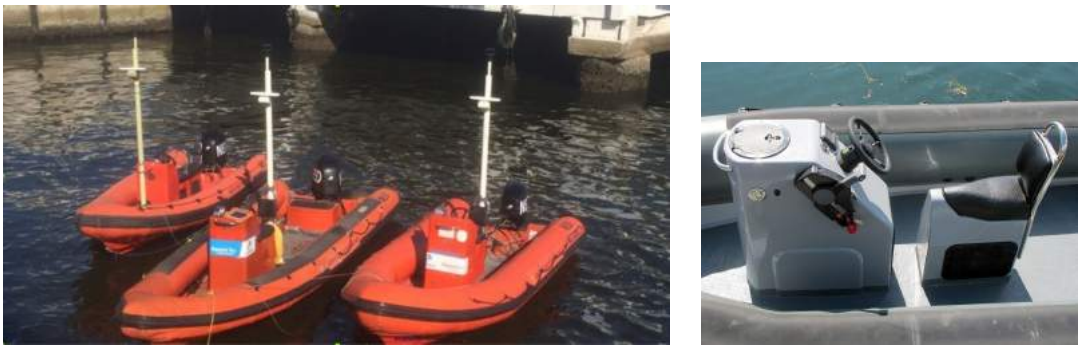


FIGURE 2.5: Example of a Rigid-Inflatable Boat (RIB) and Console closeup. (Source: NTCZ and Tornado Boats)

Chapter 3

Prior Work

In this chapter prior work related to this thesis will be reviewed. First, we will review the current situation of how video is captured and used now in sailing at the Nationaal Topsportcentrum Zeilen (NTCZ). To the best of our knowledge there has been no previous research on (semi-)automated video analysis in sailing for coaching. However, there is some relevant work where video is used in the context of sailing as well as video analysis that is being used in other sports. This will be reviewed in Section 3.2. Next to this, there are multiple aspects in this thesis that have relevant prior work. Relevant stabilization work will be reviewed in Section 3.3. For automatic manoeuvre detection from video there is again no published work, to the best of our knowledge. Therefore, relevant related practices and work will be reviewed in Section 3.4.

3.1 Current situation

As is common in many other sports, the coaches of the Dutch sailing team make use of video analysis in their coaching. However, how the footage is acquired and used right now varies from coach to coach. What is true for most coaches is the fact that they manually record short videos whenever they see or expect to see something interesting that they could use in their coaching. In Figure 3.1 a schematic representation of the process can be found. The coach follows a boat and sees something relevant to record (Figure 3.1, Step 1). They take their recording device and record a video (Step 2). The process of recording these videos usually requires one and sometimes two hands. This is an undesirable situation as the coaches do actually need to use their hands to operate the RIB, as described earlier in Section 2.1.4. Some of the coaches record videos using their phones where others use a handheld camera. The videos that were recorded by the coaches are then used, for example, in the debrief after a training session. It depends on the coach if the videos are organized and stored in some database, sent to the sailors or just kept on their phones or cameras (Step 3).

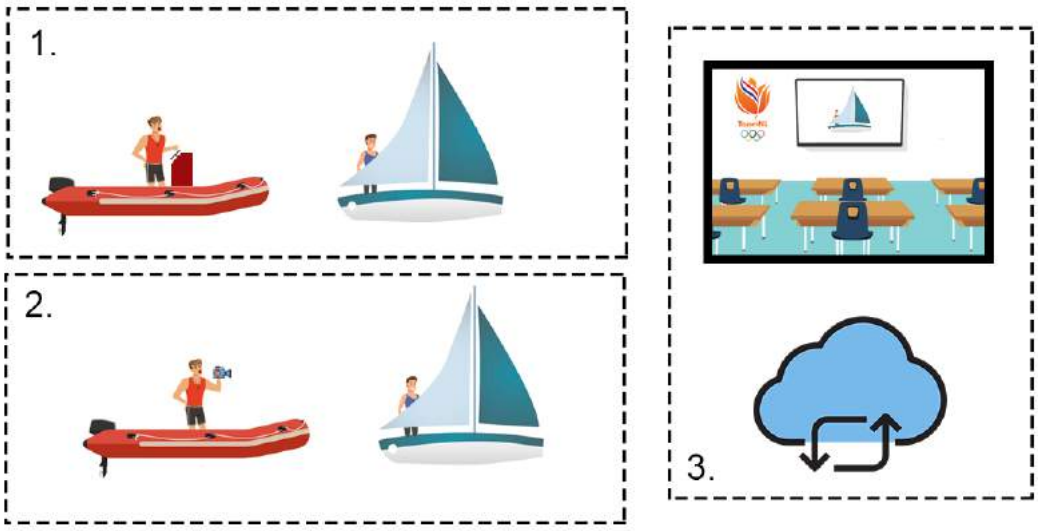


FIGURE 3.1: The process of capturing video clips by coaches. (Step 1. Follow sailor, Step 2. take recording device and record and Step 3. Share/Show videos.)

New situation In the nearby future the coaches of the Dutch Olympic sailing team should all have a camera mounted to a pole that will continuously record videos and a large button on top of the RIB console they can press to mark a point in time whenever they observe something interesting (see Figure 3.2, Image 2). This way the coaches have their hands free to operate the RIB safely and can quickly press the button when they want to.

Tests with the described setup are ongoing and are provided by the company annalisa. The new system will be available to the coaches in the near future and is expected to alleviate the burden of recording using their phones or other devices. In addition, by recording the whole session no events will be missed.

The downside of mounting the camera to a pole on the coach RIB is that the recorded videos will be less stable. The motion of the RIB is now translated directly to the motion of the video camera potentially causing the recorded videos to be unstable. In the situation where a coach holds a camera or phone to record, some of the motion would be partly compensated by the human holding the recording device.

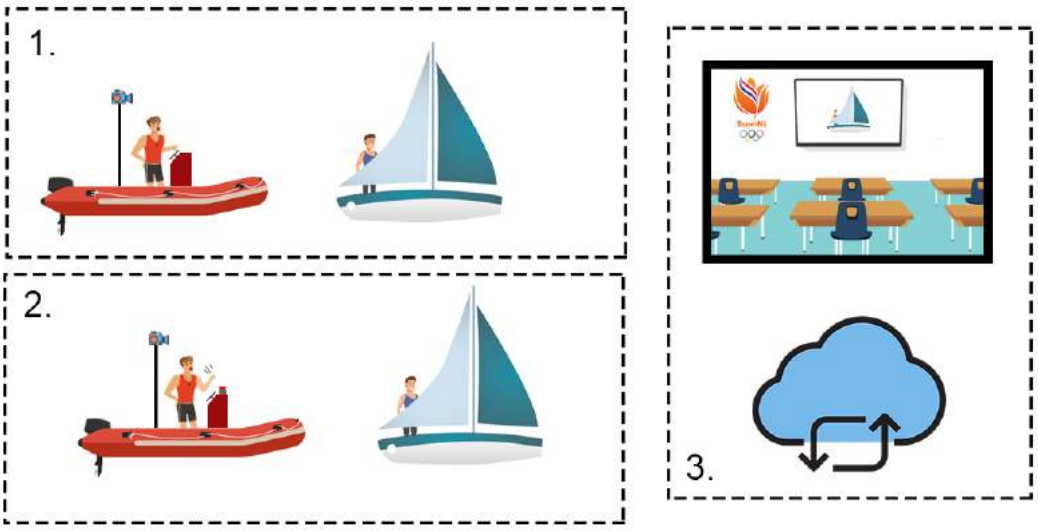


FIGURE 3.2: Coach follows boat with camera mounted and always recording. Red button can be pressed to mark interesting moments.

3.2 Video analysis in sailing

Since video cameras have been around for quite a while now it is no wonder that video is often used as a tool in coaching across many different sports. These videos can be used to provide feedback to athletes, as described in the work of O'Donoghue [6]. Research focused on video analysis in sailing is, unfortunately, scarcely available. However, there are a number of publications that make use of video, such as for example for the analysis during the development and operation of sailing simulators [7], [8], [9].

Video analysis in sailing has also been used before to study technique. To study movement behaviour a camera was mounted on the bow of a Laser [10]. From the videos captured by this camera the heel angle was determined using a computer screen, to display and pause the video, and a protractor, to measure the angle between the mast and the horizon. Temporal patterns and the nature of physical activities were studied by analysing video recordings of simulated races [11]. Mackie [12] reported on the development of a protocol to assess hiking technique using video.

Although published research on performance monitoring systems in dinghy sailing is rare, Boehm et al. presented such a monitoring system[13]. Besides a number of sensors the system also included audio and video streams to monitor the crew and observe the sails. This system should allow coaches and sailors to discuss and evaluate performance based on the sensor and video data. During the Americas Cup another monitoring system was used to gather data, which was then used to augment this data on the video broadcast [14]. This system was, however, focused on making the TV broadcast more attractive and understandable for viewers. But, it shows that measuring and augmenting data can be helpful in analyzing and understanding what is going on in the videos, which in turn could potentially be helpful for coaching in sailing.

3.3 Stabilization based on Horizon line

Over the years cameras have become more affordable and able to capture footage of higher quality. This leads to cameras being used to replace other sensors or combinations of sensors. For example, using the detected horizon line as an alternative for inertial sensors in unmanned aerial vehicles [15].

This same idea, detecting the horizon, can instead of navigation also be used to stabilize the captured images by the camera. Several similar approaches, that detect and utilize the horizon line to stabilize, are available [16], [17], [18]. These approaches aim to detect the line in the image sequence using edge detection algorithms or other methods and pre-processing steps.

Another approach is using a Support-Vector Machine (SVM) to separate a binary image into two regions, and, using this to find the horizon [19]. In line with the SVM approach is the pixel-wise segmentation approach using a Fully Convolutional Network [20]. These approaches are generally computationally expensive but can achieve high accuracy.

Other examples of approaches include the use of features detected around the horizon [21], corner points detected by an adaptive Harris algorithm [22] and hybrid approaches using a features and dense method [23].

3.4 Manoeuvre Detection

Manoeuvre detection has been done before using the apparent wind angle [13]. This approach requires sensors to measure the wind angle but does give a reliable estimation of when a manoeuvre happened. Unfortunately, these sensors are not usually found on dinghy sailing boats but if available could prove to be a valuable addition.

Although examples of research exist in the automotive domain where video data is used to detect or estimate manoeuvres [24], this does not hold for sailing. Especially when looking for automatic manoeuvre detection based solely on video data there is no published research, to the best of our knowledge.

Therefore, on a higher level we will look at vision based detection and tracking for video analysis in other sports. In addition, we will also review object detection and tracking which relates to detecting manoeuvres based solely on video data.

To provide coach assistance and performance achievement automatic body tracking and motion analysis has been used [25], [26]. Other approaches include, but are not limited to, event tactic analysis [27], team tracking techniques [28] and vision-based systems aimed at soccer videos [29]. These methods are examples which are related to manoeuvre detection in the sense that we need to derive information from the video to be able to detect the manoeuvres. In this thesis this means the need to detect the boat and sailors.

Object detection and tracking have become a relatively popular area of research nowadays. There is a vast amount of work available on object tracking, of which overviews are given in surveys [30], [31]. This holds for object detection as well, see for example the survey by Parekh, Thakore, and Jaliya [32]. Lately we have seen an increasing interest in deep learning methods for object detection [33]. Next to this, other interesting work is the moving object detection survey by Yazdi and Bouwmans [34], which focuses on methods for detecting moving objects from a moving camera.

Chapter 4

Method

4.1 Overview

To be able to find, extract and visualize segments we need to go through a number of steps in what we will refer to as the pipeline. An overview of the pipeline is illustrated in Figure 4.1. First we will briefly address the motivation and purpose of each step in the pipeline. The sections that follow (Section 4.2 - 4.6) in this chapter will explain the methods used in each step in detail.

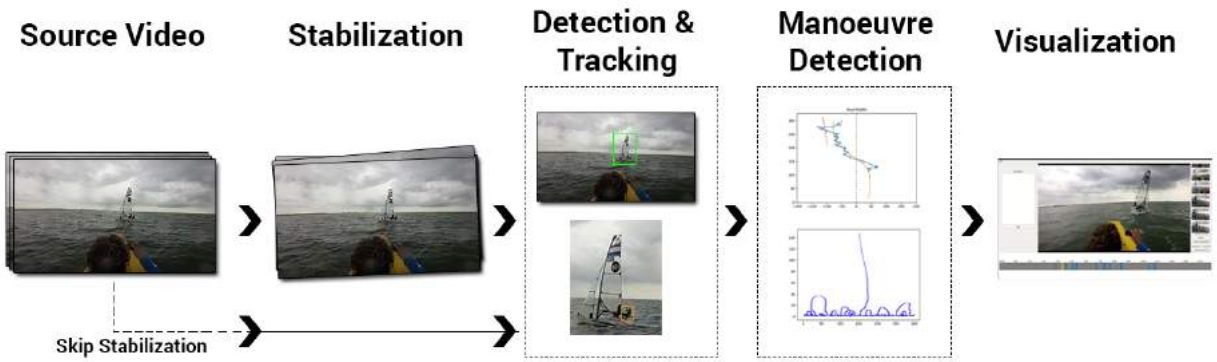


FIGURE 4.1: High level overview of the pipeline.

Source Video With the camera mounted to a RIB a whole training session can be recorded, as described earlier in Section 3.1. Therefore there will be no manual compensation of the RIB motion as before, when the coach held the camera.

Stabilization To compensate the induced motion source videos will be stabilized. Our hypothesis is that by stabilizing the video it becomes easier to analyze for coaching purposes, compared to a non-stabilized version of the same video. Next to this, we assume the stabilized videos will lead to an improved performance of the next step in the pipeline, detecting and tracking the sailing boat and persons. Although the stabilization can be skipped, the Detection and Tracking could benefit from the stabilized footage, as the reduced motion of the object of interest should make it easier to follow. This is of less importance for the Detection, which does not use time-coherent data but only uses separate frames, but our hypothesis is that stabilizing the videos makes a real difference for the Tracking.

Detection and Tracking Manoeuvres are an important aspect of sailing and are therefore considered to be interesting to review in recorded videos. Manoeuvres in this thesis refer to Tacking and Jibing, as described in Section 2.1.2. To be able to determine when a manoeuvre happened from the video data, we need to detect and track both the location of the boat and the location of the sailors with respect to the boat. Our assumption is that the RIB is following the sailing boat from behind. Nonetheless, tracking the location of the boat in the video provides information about whether there is a boat in view or not, and this information can be used to label time intervals where there is no boat in view as unimportant.

Manoeuvre Detection With the boat and sailor locations known we can try to detect manoeuvres. As manoeuvres are an important part of sailing, building an archive of manoeuvre videos can yield interesting insights. With the points in time where manoeuvres happened known you could label all the manoeuvres of a training session as interesting intervals in time. These intervals, can then be used in the debrief after a training session and stored to create an archive of manoeuvres. A potential future use of this archive is to annotate the archived data and use it as a training set for machine learning algorithms. This would be especially interesting when combining the video data with other sensor data, but will not be in the scope of this thesis.

Visualization The last step of the pipeline is visualizing (parts of) the previously extracted information. Going through the numbers and intervals by hand would be a cumbersome and time-consuming task, which can be improved by using the visual pipeline in a framework. This visual analysis framework should allow the coaches and sailors to easily search through a recorded training session and highlight the intervals that will most likely interest them. This also provides an opportunity to enrich the video data using, for example, annotations.

4.2 Stabilization

The goal is to stabilize the video to make the footage of the boat easier to analyze. To do so, we need to compensate the motion of the RIB. Because we are only using video data we need a visual cue which we can use to calculate the transformations needed to compensate the motion of the camera. A promising visual cue in this case is the horizon. We assume that the horizon is always visible in a marine environment, as it is very likely that there will be a visible edge where the water stops and transitions to sky or background. Next to this, we assume that the horizon is a relatively long and more or less straight line.

An overview of the steps of the video stabilization pipeline can be found in Figure 4.2. The idea is to locate the horizon line, our visual cue, in the video frame using an edge detector. We run the edge detector and find lines in the video frame, and from these lines we need to select the horizon. The detected horizon line is then used to calculate the transformations to stabilize the video. Next, we describe the subsequent steps of the pipeline in details.

4.2.1 Pre-Processing

We start with the first frame of the video. If this frame is not already in grayscale we convert it to grayscale. We could have used each color channel separately, but

FIGURE 4.2: Visual representation of the Stabilization pipeline.

this would complicate the model and we assume it will not significantly improve the detection of the horizon line, as 90% of the detected edge pixels in grayscale images are the same for color images [35]. Based on this assumption, we decided to use grayscale images for the edge detector.

Converting to Grayscale To convert to grayscale we rely on recommendation BT.601 by the International Telecommunication Union – Radiocommunication [36]. The luminance (E'_Y) is calculated according to the prescribed formula:

$$E'_Y = 0.299E'_R + 0.587E'_G + 0.114E'_B$$

Converting the original frame as in Figure 4.3a using the formula above results in the image in Figure 4.3b.

(A) Original video frame.

(B) Original video frame converted to grayscale.

FIGURE 4.3: Conversion to grayscale.

Median Blur To prepare the frame for the edge detector algorithm, we remove noise by applying a median blur. The median blur, introduced by Turkey in 1977 as stated in the work of Weiss [37], was selected because it is edge preserving and has just one parameter, the kernel size. A kernel size of 7x7 was the smallest kernel that, empirically, gave good results given the available relevant test data. See Figures 4.4a and 4.4b for an example of an input frame and the denoised result. The difference might be hard to observe in this document, but the best observable difference can be seen in the sail of the sailing boat. However, what is more important is the difference the denoised image makes for the output of the edge-detector. Applying the median blur reduces the number of edges detected on waves using the edge-detector. This difference can clearly be observed after applying the Edge Detector algorithm in Figures 4.5 and 4.6.

(A) Input video frame.

(B) Denoised frame using median blur.

FIGURE 4.4: Denoising video frame using Median Blur.

4.2.2 Canny Edge Detector

With the frame converted to grayscale and denoised the next stage in the pipeline is the detection of edges. This is achieved by applying the Canny Edge Detection algorithm [38] to the pre-processed frame. The hysteresis thresholds for the edge detection algorithm are set to $(50, 125)$. An example of the output of the Edge Detection algorithm can be found in Figure 4.6, when the same frame is not denoised first we get an image such as in Figure 4.5.

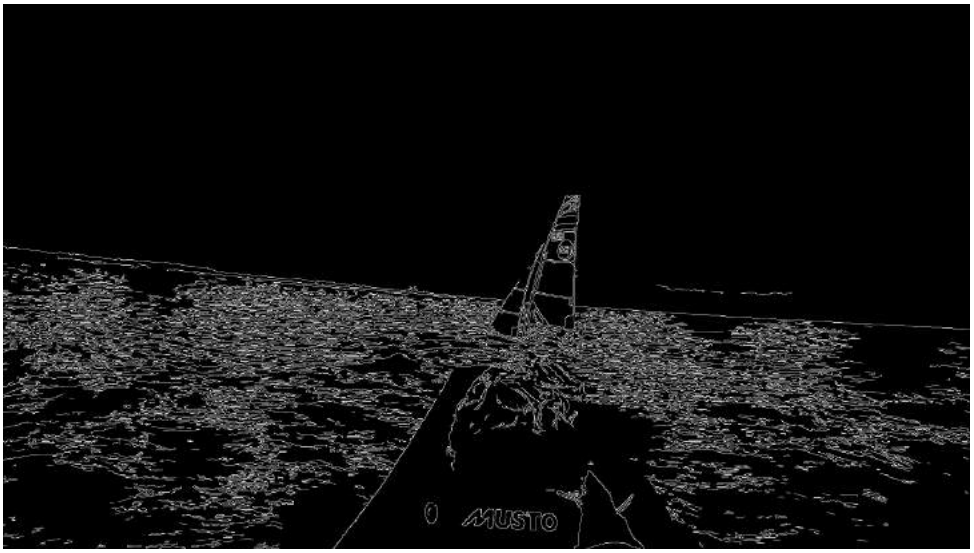


FIGURE 4.5: Output of Canny Edge Detection without denoising the frame.

FIGURE 4.6: Output of Canny Edge Detection from pipeline after denoising.

4.2.3 Horizon Line Selection

The Horizon Line Selection consists of three steps. First, we prepare the edges for the line detection method using dilation. Next, we use a line detection algorithm to retrieve a set of lines. This set is then sorted and the best candidate for the horizon line is selected.

Dilation To increase the probability of detecting the horizon, the output of the Canny Edge Detection algorithm is dilated. The horizon line detection is based on a voting system, and using dilation helps to make the horizon line a more prominent continuous line, thereby increasing the probability that it will be selected.

A 3x3 kernel containing ones is constructed which represents the neighbourhood of pixels over which the maximum is taken when moving the kernel over the image. For an example of the operation, using a 3x3 kernel as structuring element, see Figure 4.7. As an example, after two iterations of the dilation operation over the image in Figure 4.6 we achieve the result in Figure 4.8.

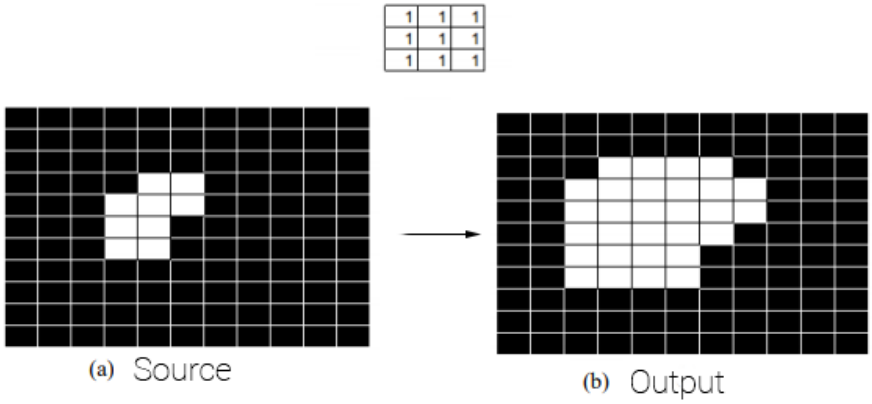


FIGURE 4.7: Dilated operation using 3x3 structuring element. Image modified from original [39].

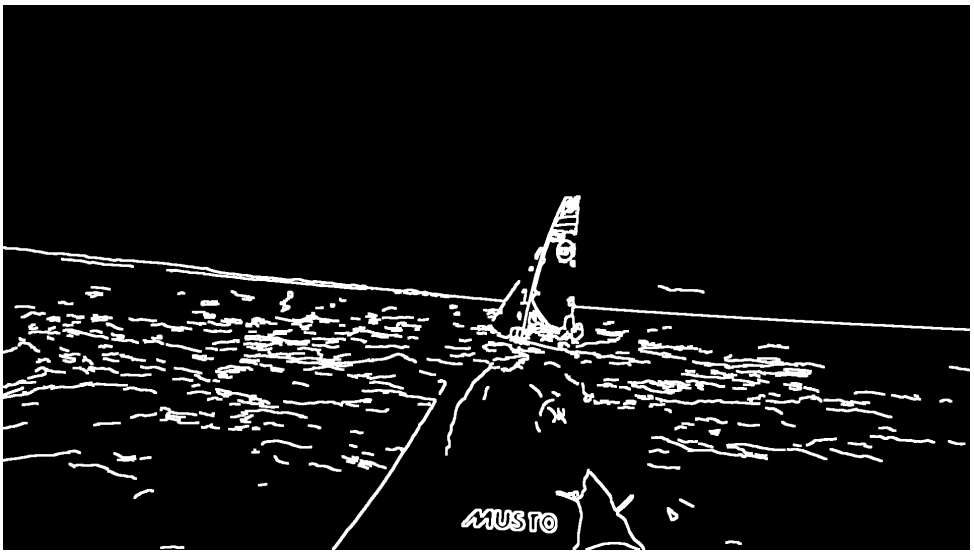


FIGURE 4.8: Dilated output of Canny Edge Detection.

Line segment detection Given the dilated edges, the next step is detecting and extracting line segments in the binary image. A well known and robust method for line detection is the Hough Transform, a technique to find geometric primitives by using a voting procedure in a parameter space. A downside of the original Hough Transform algorithm is that it is computationally expensive. However, using the progressive probabilistic Hough transform (PPHT) algorithm [40], line segments are detected in the source image with less computations. The progressive probabilistic Hough Transform differs from the standard Hough Transform by repeatedly selecting a random point for voting. When a bin exceeds the voting threshold, allowing a decision, we have detected a line and can remove the supporting points. Points that remain, which support the detected line, will be removed from the points that have not voted and continues with another randomly selected point. The parameters for the PPHT are set to the following: $\rho = 1, \theta = 0.01$, which are the same values used in the original paper [40]. The voting threshold during experiments was set to 150 votes. Next to this, a minimum line segment length of 100 pixels was used during

the experiments to avoid selecting short lines on the water surface. Note that the minimum line length has to be accounted for when using low resolution videos. For an example output of the PPHT with these parameters see Figure 4.9a.

(A) Detected line segments using PPHT.

(B) Longest detected line of PPHT, the horizon.

FIGURE 4.9: Visualization of the output of the Progressive Probabilistic Hough Transform and the line selected from the set.

Line Selection The output of the PPHT algorithm is a set of line segments detected in the frame. We assume that the longest horizontal line segment in this set is located on the visible horizon. Therefore, we sort the set of line segments (defined by a start point and end point with x and y coordinate) by the absolute difference between the x coordinates of the start and end points of a line. The line segment with the largest absolute difference in the x direction is selected.

4.2.4 Transformation

The transformations are calculated using the horizon detected using the processing pipeline and the artificial target horizon as depicted in Figure 4.10. Note that the bottom and top part of the image are darkened, this is merely for visualization purposes in this thesis and serves no use in the pipeline. The translations and rotation that transform the detected line to the destination are calculated and stacked in a data structure. This stack is then filtered using a moving average filter and the transformations applied to the corresponding original frames.

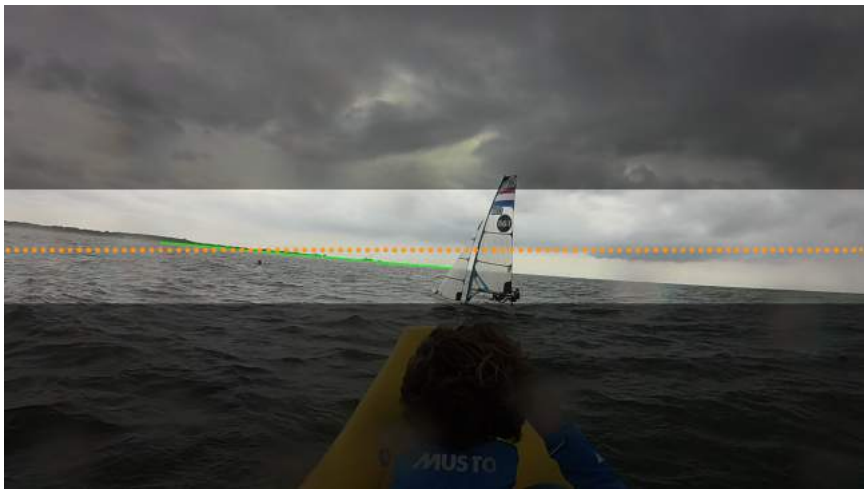
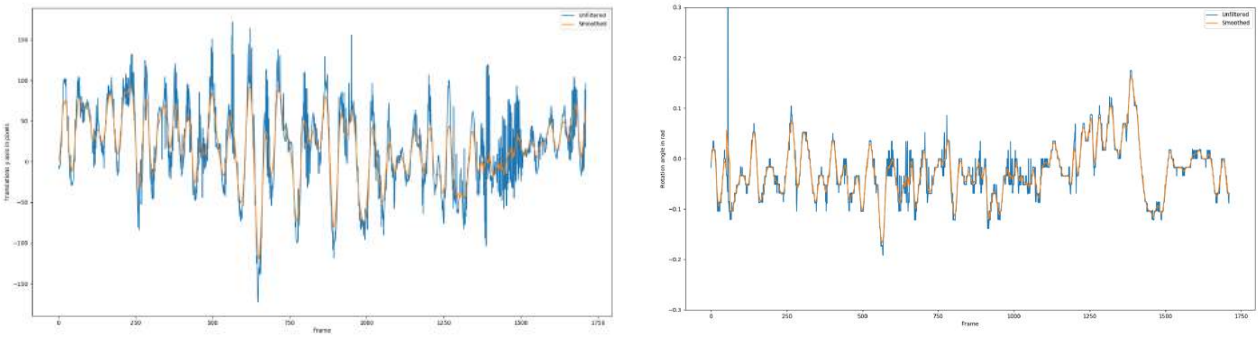


FIGURE 4.10: Detected horizon (Green) and destination (Orange dotted). Top and bottom part darkened for visualization purposes.

The graphs in Figures 4.11a and 4.11b represent the calculated transformations that will be applied to the original video to end up with the stabilized result. The smoothed transformations, the orange line in the graphs that is created by averaging the values within a window in time, are the transformations that will be used. This results in a smoother transition between sequential video frames. More examples, similar to Figure 4.11, can be found in Appendix A.



(A) Translation in y axis per frame, Unfiltered (Blue) and Smoothed (Orange).

(B) Rotational angle in radians per frame, Unfiltered (Blue) and Smoothed (Orange).

FIGURE 4.11: Video stabilization graps of transformations.

By applying the transformations to the frame we translate and rotate the frames. These transformations introduce black areas around the edges, see Figure 4.12b. The black area in the images gives an indication of how much the frame is rotated and translated. If desired, these areas can be filled using inpainting techniques. For example, using one of the inpainting techniques reviewed by Qureshi, Deriche, Beghdadi, *et al.* [41]. The black areas can also be removed by cropping the frame if the resolution and field of view of the camera are suitable for this. However, both approaches can cause problems. Cropping could remove important visual information and inpainting may not be perfect and therefore still be distracting. Therefore we will not apply these methods but will evaluate in Chapter 5 if the black areas are distracting for the users. We leave solving this, if the evaluation indicates this is a problem, for future work.

Next to this, in Chapter 5 we will also evaluate if the assumption (see Section 4.1) that the stabilization will improve the Detection and Tracking holds. The Detection and Tracking will be explained in Sections 4.3 and 4.4.

(A) Original video frame. (B) Frame after applying transformation.

FIGURE 4.12: Transforming the video frames causing black areas around edges.

4.2.5 Limitations

Because the stabilization is based on detecting the horizon this introduces limitations on the method. Whenever there is a long high contrast line that is not parallel to the horizon it could be detected as being the horizon and in turn stabilize the video frames with respect to this line. Although the probability of encountering such a situation is low and during the experiments it was not encountered, it could still occur and therefore limit the stabilization performance. Imagine for example a rope or beam in the field of view of the camera or a large object close by obscuring most of or the entire view.

Another limitation comes from the assumption that the horizon is more or less a straight line. Because of geometric distortion, often introduced by a wide-angle lens, the horizon can be a curve in some situations. This limitation could be overcome by compensating the geometric distortion using for example one of the methods reviewed by Hughes, Glavin, Jones, *et al.* [42]. It does however require the camera to be calibrated and we will leave this as future work.

Next to this, in case there is heavy rain, the raindrops on the lens of the camera can distort the image too much resulting in the stabilization method not working properly. The performance under these conditions can be improved upon by equalizing the grayscale histogram and increasing the contrast around the horizon, or by avoiding rain drops hitting the lens of the camera in the first place. Videos recorded with a lens covered in raindrops is most likely not going to be of any use anyway.

4.2.6 Computational Optimizations

Several computational optimizations could be added to improve the processing speed of the pipeline. One of these optimizations is downscaling the resolution of the source frames, which means there are less pixels to process and is therefore faster. Of course, by reducing the resolution the amount of information (in the form of pixels) is also reduced. However, downscaling the resolution can aid the correct and faster detection of the horizon because there will be less lines detected by the PPHT. An example of this can be seen in Figure 4.13 (original frame in Figure 4.14a, where in the downscaled version of the frame there is less noise on the water and around the horizon in the Canny edge detection output. Downscaling too much can however become problematic and cause the pipeline to not produce any results. When we

downscale the original frame to 480x230 pixels the pipeline does not work anymore when keeping the parameters the same. This is because the dilated detected edges will then clog most of the image making the detection of the horizon line difficult in most cases. These parameters can be tuned to be able to work with low resolution videos, but if the trade-off between processing speed and stability is worth it depends on the use case. Since we are not aiming for a real-time performance, this was not the case here.

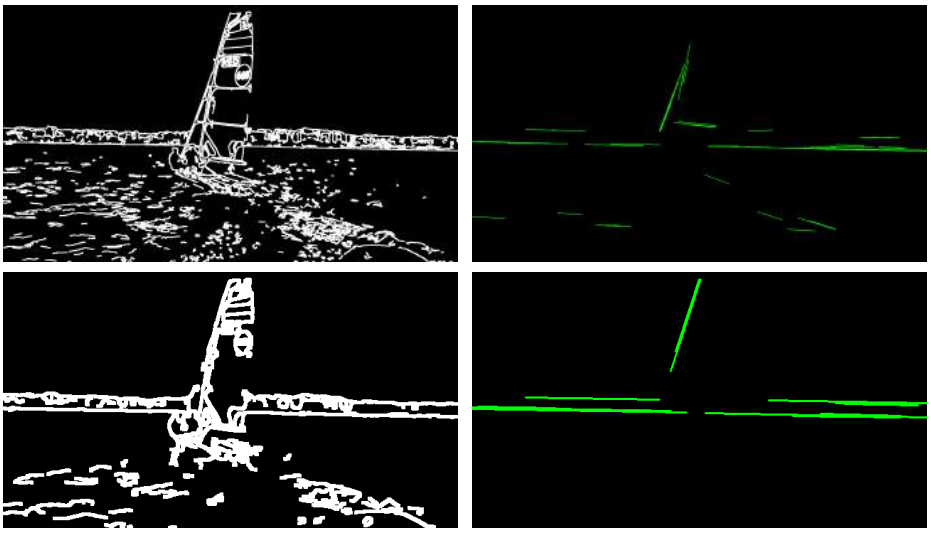


FIGURE 4.13: Dilated Canny output and detected lines of a 1920x1080 pixel frame (top row) and a downscaled version of 640x430 pixels (bottom row)

Another possible optimization is cropping the source video frames around the horizon. By cropping away for example the bottom and top 20% of the pixels, as can be seen in Figure 4.14b we reduce the data that needs to be processed to calculate the transformations. To achieve this a full frame is processed to detect the horizon in this frame. The subsequent frame is then cropped around the horizon that was detected in the previous frame. The assumption here is that the horizon will not shift out of the range over the course of 1 frame. Whenever the horizon is not detected one would have to fallback to using the full frame. Small scale experiments with cropping 20% of top and bottom resulted in a speedup of around 2.

(A) Original unprocessed video frame. (B) Cropped around horizon.

FIGURE 4.14: Potential stabilization method speedup by reducing the amount of pixels processed.

4.3 Detection

As mentioned before in Section 4.1, manoeuvres are an important part of sailing. To detect these manoeuvres using the video data we want to know where the boat is and where the sailors are located in relation to the boat. Over the last couple of years we have seen a steady increase in popularity of Deep Learning methods for Object Detection. This is supported in the review by Zhao, Zheng, Xu, *et al.* [43], which also mentions the advantages of Deep Learning over traditional architectures. Next to this, using a Deep Learning network should allow for detecting boats and persons in a wide variety of situations and environments with relatively high accuracy. The Deep Learning network that we chose to use in this thesis is the MobileNets Single Shot Detector Convolutional Neural Network [44].

MobileNets We selected MobileNet because it is an efficient model, originally designed by Google as a light weight deep neural network that could be used on mobile devices [44] [45]. Moreover, Wu, Sahoo, and Hoi stated that without significant loss in accuracy Mobilenet significantly reduced computation cost and at the same time reduced the number of parameters. Next to this, our decision was strenghtened by the fact that a pre-trained model is readily available, relieving us of the burden of training the network [1]. This model was trained on the Microsoft Common Objects in COntext (COCO) dataset [47], which contains images for the classes *Person* and *Boat*. Although this model is not specifically trained on dinghy sailing boats and the sailors on it, we assume that detection performance of persons and boats in video frames will be sufficient. We leave training the network on a specific dinghy sailing boat dataset for future work, in part because we do not yet have a sufficient dataset available to do this.

With the previously mentioned deep learning network we can detect the boat and persons by feeding the video frames to the network. However, we first need to prepare the video frames. For a schematic overview of the steps see Figure 4.15. In the sections below the steps of this pipeline will be discussed in more detail.

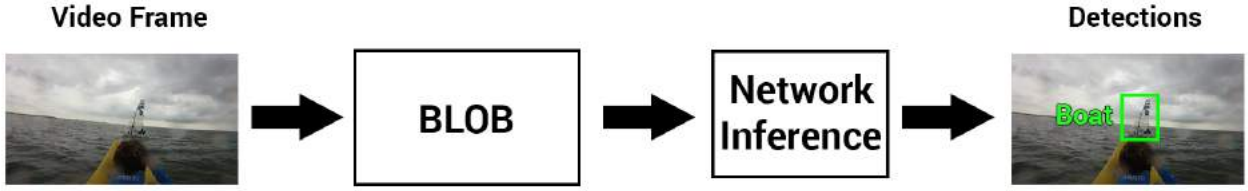


FIGURE 4.15: Schematic overview of the pipeline used to detect boats and persons in video frames.

4.3.1 Preparing the image for Network Inference

Because the deep learning network we use was trained using 300x300 pixel "blobs" (4D tensors consisting of images, channels, width and height) from images we have to pre-process the images. This process entails mean subtraction and scaling the image. Because most of the video data we have available is of a higher resolution

[1] https://github.com/chuanqi305/MobileNet-SSD/

than 300x300 pixels, there is a need to down-scale most of the input. However, we down-scale the images to a height of 300 pixels while maintaining the same aspect ratio as the input images. For an example of a pre-processed frame see Figure 4.16.

As an experiment we compared down-scaling to different resolutions from a source video of 1920x1080 pixels. This was used as a sanity check. It confirmed the network did indeed perform better, where performance in this case refers to the boat being detected in the frame, when the images were down-scaled to a height of 300 pixels (see Figure 4.17). Additionally, we tracked the middle of the boat manually for this video and plotted the x coordinate. The middle x-coordinate was determined by taking the middle of the left and right edge of the bounding box, measured in pixels (pixel 0 on the x-axis is the leftmost column of pixels in the image). The middle x coordinate of the manually tracked boat and the 533x300 down-scaled images follow more or less the same trajectory, albeit with an offset caused by the manual selection of the middle of the boat.

FIGURE 4.16: The "blob", obtained by pre-processing a video frame. The three color channels (Red, Green and Blue) are shown.

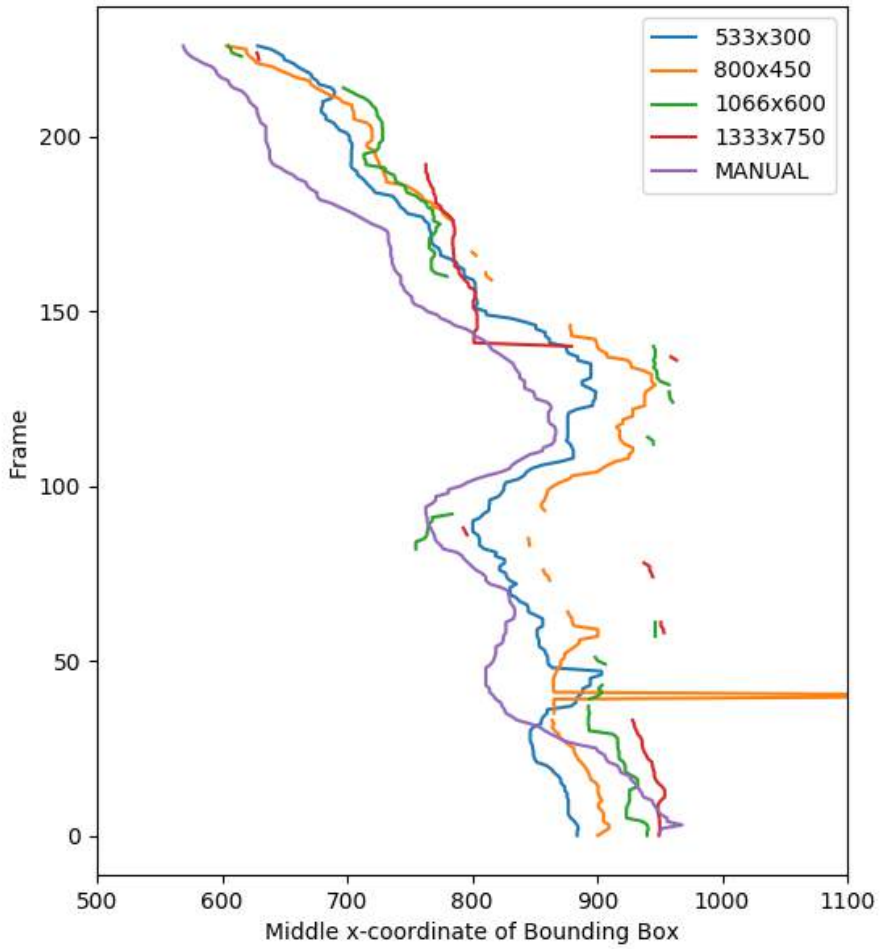
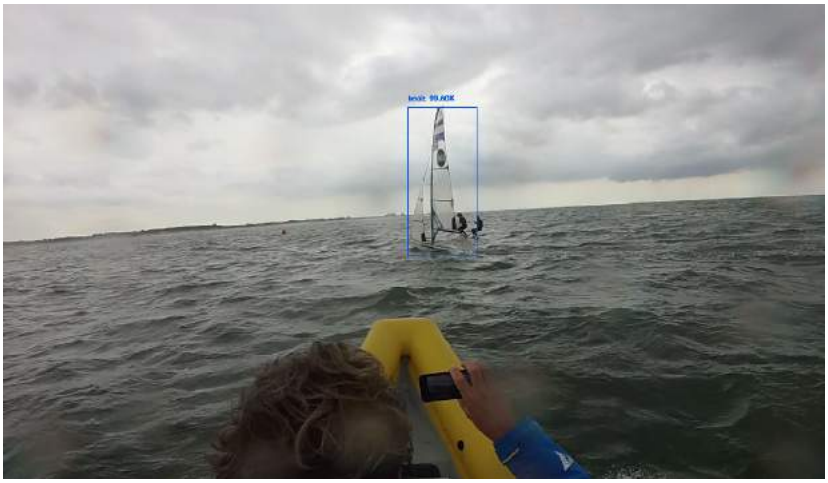


FIGURE 4.17: Graph of the difference in detection of the boat in the video frames when downscaling to different resolutions. Also contains the manually tracked middle of the boat.

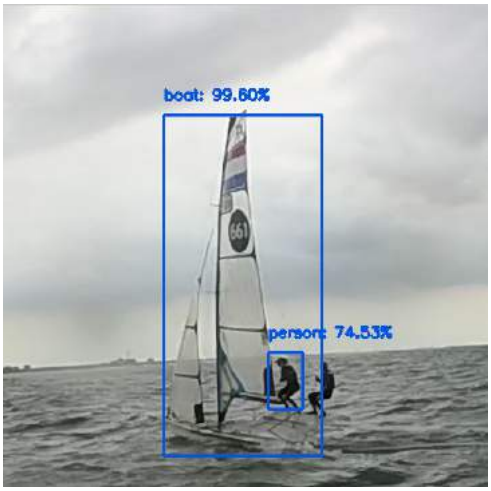
4.3.2 Network Inference and Detection

With the "blob" prepared, we feed it to the network to detect the boat and sailors. As a result, the network will return a bounding box around a detected object, together with the class and confidence associated with that bounding box. For an example, see Figure 4.18a. All bounding boxes with a confidence lower than 66% are discarded to filter out noise. However, the network could still return multiple bounding boxes surrounding the same object. To end up with just one bounding box around an object we apply a technique called Non-Maximum Suppression. We applied the same technique as Malisiewicz, Gupta, and Efros which used non maximum-suppresion in their work [48] and made the code publicly available[2]. Using this algorithm we calculate the overlap ratio between bounding boxes and only keep the bounding boxes that do not cross the set threshold. The threshold used in this case was an overlap ratio of 0.5.

[2] `http://www.cs.cmu.edu/~tmalisie/projects/iccv11/index.html`



(A) Boat bounding box detected in video frame with a confidence of 99.60%.



(B) Person detected within the bounding box of the boat with a confidence of 74.35%.

FIGURE 4.18: Detection of boat and person in video frame.

We have separated the detection of boat and person into two steps. By this we mean that we first feed the entire prepared video frame to the network and try to detect the boat. Next, we use the bounding box of the detected boat and crop the video frame to the bounding box around the boat. After the first frame this will be done using the tracking method, this will be discussed in Section 4.4. Then, the cropped video frame is again fed to the network to detect the sailor(s). An example of this can be seen in Figure 4.18b. This is done because of the following reasons: First, whenever there is a person on the RIB within the field of view of the recording camera we would detect this person even though this would not be one of the sailors that we are interested in. Although we could mask out the RIB part of the video frame, we opted to take another approach instead. Masking the RIB would introduce more parameters and methods, as the RIBs vary in size, color and mounting location and angle of the camera. Therefore, to avoid testing and tuning extra methods and parameters we separated the detection in two steps. Second, during experiments the detection rate of persons was, empirically, quite low. This was mainly caused by the image being resized for the neural network, making the persons in the frame too small to be recognized as such. By feeding the cropped video frame to the network to detect the persons this problem was reduced, because the image would not have to be resized as much compared to the original video frame.

Unfortunately, during experiments it became apparent that the detection did not always continuously detect the boat when it was present in the video frames. For some videos we can detect the boat in every frame using the deep learning network, see for example the “533x300” line in Figure 4.17. However, this does not hold for all videos. For some videos, for example when the recording camera is further away or when recording the boat from an angle that makes it difficult to recognize a boat as such, there are large gaps in time where the boat was not detected for multiple sequential frames (see Figure 4.19). When the camera is too far away there is not enough resolution to detect the boat.

As mentioned previously, the cropped video is used as input to detect the persons on the boat. During experiments we noticed that the detection becomes too unstable below a resolution a 200x200 pixels when trying to detect persons on the

boat. Using this resolution as a minimum threshold, empirically, produced good results. When the detected bounding box around the boat has a total resolution of less than 40,000 pixels we do not attempt to detect the persons on the boat.

One of the options would be to filter the gaps, for example using filters such as the Kalman Filter [49]. But, as we have the frames available, we can leverage this data by using the detections of the network as input for a tracking algorithm. This will be discussed in more detail in Section 4.4.

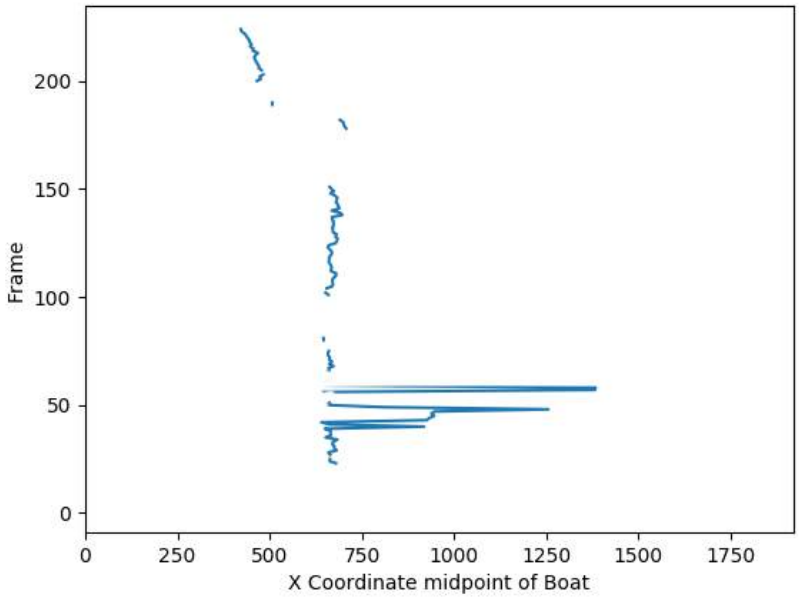


FIGURE 4.19: Graph of x coordinate of middle of bounding box detected per frame, measured in pixels and the x-axis being the horizontal axis of the frame.

4.4 Tracking

As mentioned before in Section 4.3, we can use the detection output of the network as input for a tracking method. This allows for automatically initializing a tracker which we assume will improve the continuous localization of the boat and sailors. By using a tracking method we make use of the temporal coherence of the sequential video frames, which was not used by the detection network. When the detection fails, the tracking method could still be tracking the object and fill in the gaps. Next to this, for most tracking methods it holds that these are faster in terms of FPS than using detection methods.

Tracking Algorithm The tracking method that was selected is the Discriminative Correlation Filter with Channel and Spatial Reliability (DCR-CSF) method [50]. This tracking method was selected because it is a relatively fast method with an excellent tracking performance. It can accurately track complex objects under rotations, occlusions and other factors while running in real-time on the CPU. The DCR-CSF approach is an extension of the Correlation Filter tracking method. The Correlation Filter works by training a filter on the appearance of an object. At the first frame the object is selected by placing a tracking window on the object of interest and training the filter. Then, we take the next frame and correlate the filter over a window in the next frame. The location in the frame where the output of the correlation filter is maximal is selected as the new location of the object. The filter is then updated using that location and the process is repeated. The Channel and Spatial Reliability extension improves the Correlation Filter. Color segmentation is used to improve which parts of the object will be used in the filter for tracking, this is referred to as the spatial reliability. The Channel Reliability refers to the calculated reliability of each feature channel used in the filter.

4.4.1 Implementation Details

The tracking algorithm is initialized using the detected bounding box around the object in the video frame. We use a separate tracker for the boat and person. The tracker is updated each frame which provides the bounding box surrounding the location of the tracked object. This information is then used in turn by the Manoeuvre Detection part of the pipeline, which will be discussed in Section 4.5

After the first frame where a boat is detected (see Section 4.3.2), the bounding box that is generated as output by the tracking algorithm is used to crop the video frame to the bounding box located around the boat. To avoid cropping the sailors off, that will be hiking and therefore outside of the boat, a margin is taken around the detected bounding box. This margin does not have to be that large, because the bounding box generated by the tracker is always larger than the bounding box of the detection step, because it is less accurate. In this thesis we used a margin of 50 pixels as, empirically, this gave good results. The location of this bounding box is averaged over 10 sequential frames to avoid jitter. We assume that within 10 frames of the video the boat will not have moved significantly and can therefore safely average over 10 frames.

Even though the tracking algorithm is relatively accurate, there will be some tracking error. This tracking error tends to accumulate over time causing the tracking to drift from the target object. To account for this tracking error we calculate the distance between the center of the detected bounding box (see Section 4.3) and the center of the bounding produced by the tracking algorithm. For a schematic

representation see Figure 4.20. If the boat is detected in a frame and the distance is higher than a set threshold, the tracker is re-initialized using the location from the detection method (see Section 4.3). The threshold used during experiments which, empirically, worked well was a distance threshold of 25 pixels.

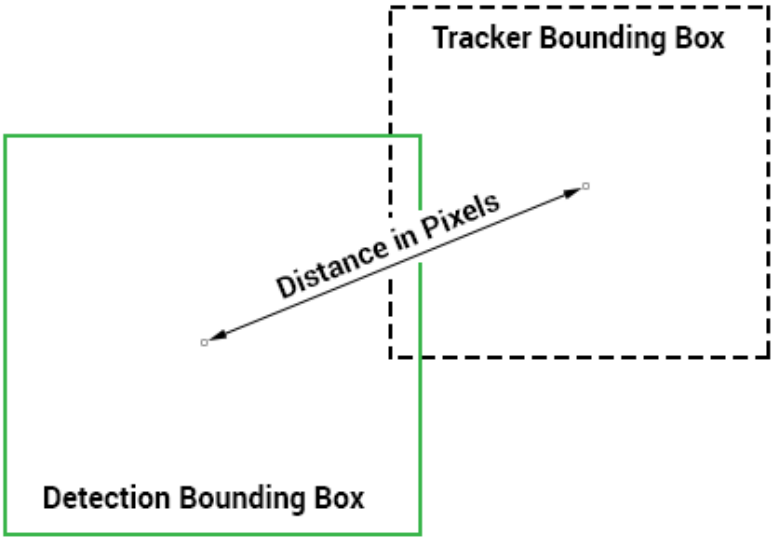


FIGURE 4.20: Calculate distance between bounding box produced by the detection method and the tracking algorithm.

4.5 Manoeuvre Detection

With the Detection and Tracking step of the pipeline we are able to detect the bounding box around the sailor and boat (see Sections 4.3 and 4.4). These bounding boxes can then be used to calculate the difference between the middle of the bounding box around the boat and person. As we are trying to detect the sailors switching sides during a manoeuvre, as described in Section 2.1.2, we only consider the horizontal difference (x axis). Even if the boat is not straight up, because of the heel angle of the boat, the sailors should still be far from the middle of the boat. Especially because the sailors will be Hiking to keep the heel angle of the boat small, which means they will be hanging outside the boat with their full body. See Figure 4.21 for a schematic representation of what the difference in this case entails. The calculated difference is measured in pixels.

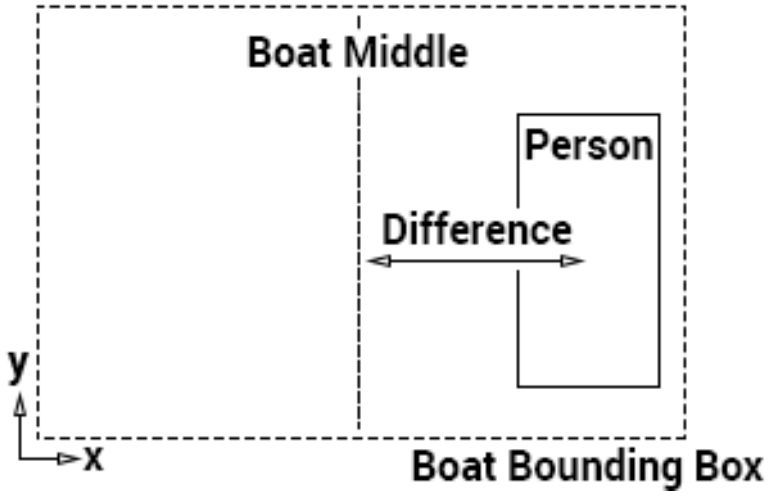


FIGURE 4.21: Calculate difference between middle of bounding boxes in x axis produced by detection method and tracking algorithm.

Because calculating the difference depends on the detection and tracking of both the boat and person, situations will occur where the difference can not be defined because one or more is not detected. Whenever the boat or person is not detected we need to deal with this missing data, as we need the difference as input for the manoeuvre detection. The approach taken to deal with this missing data is to keep the last known value up until the point where new data is available.

The idea behind this Manoeuvre Detection approach is to detect when the sailors switched to the other side of the boat. Because we assume to be following the boat from behind, we define the vertical line in the middle of the bounding box as the middle of the boat (see Boat Middle line in Figure 4.21). Whenever the sailors cross this line, and remain on the other side, we assume that a manoeuvre has just occurred. This means that we need to detect two aspects to detect the manoeuvres. First, detect on which side of the boat the sailors are. Second, detect when the sailors cross the "Boat Middle" line.

To detect the manoeuvres we combine two methods, Adapted Edge focusing and Regression Line Fitting. We use the Adapted Edge Focusing to robustly pinpoint the frame where the sailors cross the "Boat middle" line, which marks the middle of the manoeuvre. The Regression Line Fitting is used to find the start and end of the manoeuvre by fitting lines and applying a threshold to the data before and after the middle of the manoeuvre. These methods will be explained in more detail in 4.5.1 and 4.5.2.

The calculated difference data is noisy, as can be seen in Figure 4.23a where we illustrate the noisy calculated difference, Gaussian filtered calculated difference during a manoeuvre and regression line fit at the start of a manoeuvre. Because of this noisy data we will apply Adapted Edge Focusing and Regression Line Fitting to robustly detect that the sailors have moved to the other side and indeed remain there. To reliably detect the zero-crossing point, marking the middle of the manoeuvre, in the calculated difference data we would need to filter the data first. To tackle this problem we adapted a technique called Edge Focusing [51].

4.5.1 Adapted Edge Focusing

Instead of sliding a window, for which a size would need to be defined, and filtering the data in the window we use a scale space technique called Edge Focusing to find zero crossings in the noisy data. Edge-focusing is used to track edges from coarse to fine scale in different applications, for example, in medical imaging. Using the research of Bergholm [51], Witkin [52] and the description in the book by Romeny [53] (p. 221-225), an implementation of the Edge Focusing method was created and adapted to be used in this application.

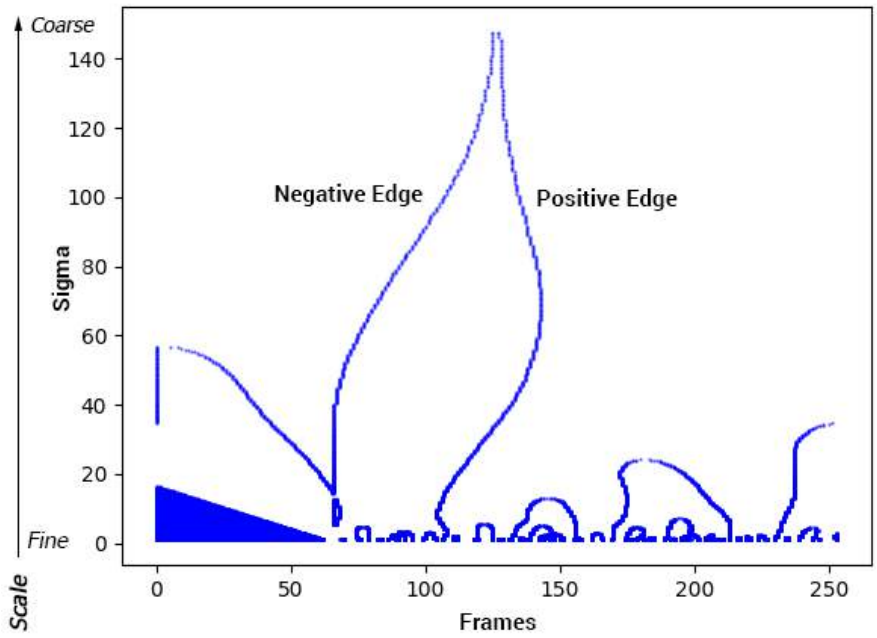


FIGURE 4.22: Adapted Edge Focusing "signature" graph. Using coarse to fine tracking following the positive edge to pinpoint zero-crossing frame.

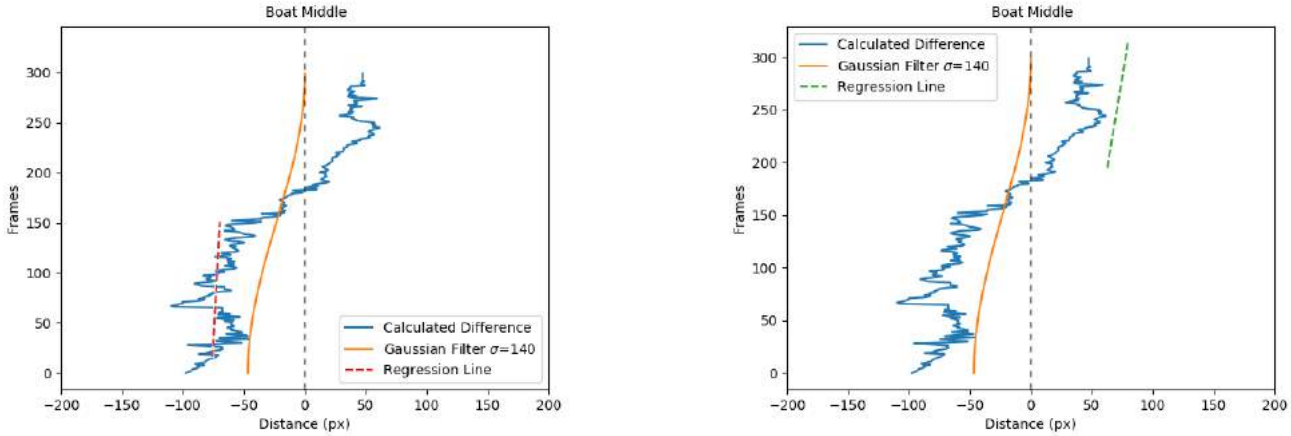
To robustly detect the zero-crossing the point is tracked from coarse to fine scale. This is done by applying a Laplacian of Gaussian (LoG) filter for a range of values for standard deviation σ. We start with a high value for σ and decrease this with small steps. In this thesis we used a range for σ of $[e^a, e^b], a = 5, b = 0$ with steps of 0.005 between 0 and 5. The stepsize of 0.005 was chosen to ensure the difference between frames forward or backward between sequential values for σ in this range is never higher than 1. We take the indices of the detected zero-crossings, which are called the "signatures", using the Laplacian of Gaussian at each value for σ in the range. When we plot all these signatures we end up with a graph like Figure 4.22. Using this collection of signatures we will track the zero-crossing from coarse to fine. By following the positive edge down from a coarse start point, for example at $\sigma = 140$, we arrive at the precise zero-crossing point at frame 110. The stepsize is necessary to be able to track the edge down, because we take steps of not more than one frame per signature when tracking the edge.

With the middle of manoeuvre determined we now turn to Regression Line Fitting to find the start and end of the manoeuvre.

4.5.2 Regression Line Fitting

We use least squares regression line fitting [54] because it is a simple and computationally relatively inexpensive method. Moreover, it is a robust way to determine if the calculated difference has stabilized without having to tune a lot of parameters. Using a window in time of 90 frames we apply simple linear regression to fit a line to the data points in this window. This window of 90 frames was chosen because the manoeuvres took, in our experiments, on average 6 seconds and the videos during the experiments were 30 frames per second. Therefore, we assume that the window of half a manoeuvre should be robust and, empirically, worked well during our experiments.

The slope of the fitted line is used to determine if the calculated difference has stabilized, meaning the sailors remain at more or less the same position on the boat. We call the line stable whenever the slope of the fitted line is lower than 0.15 radian, which empirically worked well during our experiments. This value could be increased if there is a need for tighter intervals or decreased if we want to be sure that we are really following the boat straight from behind and the sailors remain in one spot. For an example of the line fitted to the noisy data see the Regression Line in Figure 4.23a. For the frames where the line is stable we note the sign of the intercept, allowing us to differentiate between the two sides of the boat later on. With this procedure we are able to determine when the sailors remain on one side, because the fitted line is stable when this is the case. Then, to determine when a manoeuvre started and ended, we search where the line stable on one side using the slope and the sign of the intercept and wait for the line fitted to data of the frames that follow to stabilize on the other side. In Figure 4.23a we see the Regression Line fitted to a window of 90 frames with an angle below 0.15 radian, marking the start of the manoeuvre using the last frame in time in the window. In Figure 4.23b we can observe where the line stabilized on the other side, signifying the end of the manoeuvre. The output of the manoeuvre detection is a pair of frames, signifying the start and end frame of the manoeuvre. A margin can be applied to these start and end frames to capture the moments leading up to the manoeuvre in the interval.



(A) Regression line fitted to find stable point marking start of manoeuvre, before zero crossing (middle of manoeuvre).

(B) Regression line fitted to find stable point after zero crossing.

FIGURE 4.23: Fitting regression lines to calculated difference data to search for stable position of sailors on one side.

With the start, zero-crossing point and end of the manoeuvre calculated the manoeuvre intervals are now defined and serve as the output of the pipeline. These manoeuvre intervals can now be visualized together with the video as the last step and put to use by the users. The visualization method will be discussed in Section 4.6.

4.5.3 Limitations

One of the limitations comes from the assumption that the camera is following the boat from behind. Whenever this assumption does not hold, for example when the boat is more or less side to side with the RIB, it will generate false positives. This is because when we take the middle of the bounding box around the boat from the side we will see a lot of zero-crossings, as the persons and middle of the boat move in and out of frame. This in turn will trigger the manoeuvre detection and give the false positives as output. This could be dealt with by discarding these false positives manually in an interface or by making sure the RIB always follows one boat from behind.

Next to this, whenever the RIB is not following the sailing boat closely enough the persons on the boat will not be detected. This in turn will limit the ability to detect the manoeuvres.

Another limitation of this method is the inaccuracy in detecting the zero-crossings using the Adapted Edge Focusing method. In some cases the Positive Edge is tracked but leads to the wrong frame. This is often caused by missing data or very rapid changes in the calculated difference data.

4.6 Visualization

In this section we will discuss the visualization of the video and the extracted intervals using the methods in the pipeline. Visualizing the extracted manoeuvres and video is the last step of the pipeline, as can be seen in Figure 4.1. To visualize this data, which allow the coaches to perform a visual analysis, we need a visual analysis framework that is tailored to the available data. We will discuss the components of the framework and their envisioned purpose as well as the implementation details of these components.

4.6.1 Framework Components

The video, stabilized or not, can be shown “as is”, whereas the manoeuvres can benefit from a visualization approach to make it easier to find the manoeuvres and interesting intervals in the recorded training session. The main goal is to allow the coaches to analyze interesting parts of the recorded training session without the need to go through the entire recorded session. To achieve this goal we need to show the video itself and the interesting intervals. Up until this point the manoeuvres are a collection of tuples, where each tuple consists of the start and end frames of a manoeuvre. At this point in the pipeline other intervals in time, created by pushing the red button (see Section 3.1, *New Situation*), can be added to the set of intervals. These intervals will be visualized in two ways to make it easier to find the interesting intervals in the recorded session. The intervals and their relation in time with the video will be visualized using a Timeline which is illustrated schematically in figure 4.24, where we can distinguish intervals by using patterns or colors. The Video

itself will be displayed in the same view as the timeline, as can be seen in 4.24, to achieve our goal of finding the interesting intervals easily in the video. This timeline allows the coaches to not have to search through the video but can instantly select marked intervals, addressing our goal to not have to go through the entire recording. Next to this, the same time-related data will also be available in the form of a list of annotated thumbnails. This list should allow the user to easily select an interval, while the thumbnail gives a visual cue on the contents of an interval. A schematic overview of the three previously mentioned components (Video, Timeline and List) can be seen in Figure 4.24. These three components will be discussed in more detail next.

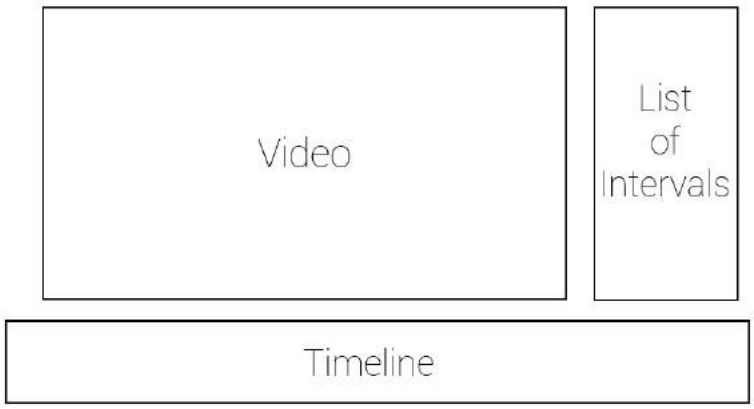


FIGURE 4.24: Schematic representation of the three important components of the visual analysis framework.

Video The video is in the end what the user will analyze. Following one of the User Interface Design principles described in the book of Galitz [55] we focus the user attention on the most important component, in this case the video. Therefore, it takes up the most space in the framework and is positioned at a central location. The video shown can either be the original video or the stabilized version of the video.

Timeline The timeline, which can be seen in Figure 4.25, is used to show the locations in time of the intervals. To allow the user to easily distinguish the intervals from the rest of the timeline we make use of color. Intervals are colored blue and the rest of the timeline a shade of gray, they can be selected by clicking on them. When clicked the interval will be highlighted using orange. These colors were chosen because orange and blue are complementary colors, and therefore should give a high contrast. When clicking the intervals the video starts playing from the beginning of the clicked interval. The user also has the option to click anywhere in the timeline and the video will start from the point in time where the cursor is located. This should allow the user to easily analyze the manoeuvres, intervals created by the red button and the rest of the video. Using the timeline the user can manually add intervals by marking the start and end of a new manoeuvre using the cursor and mark buttons.

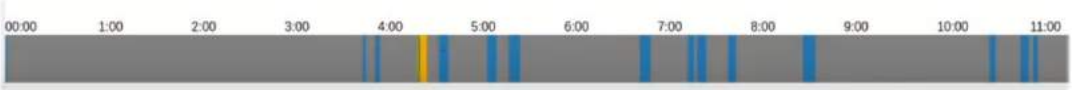
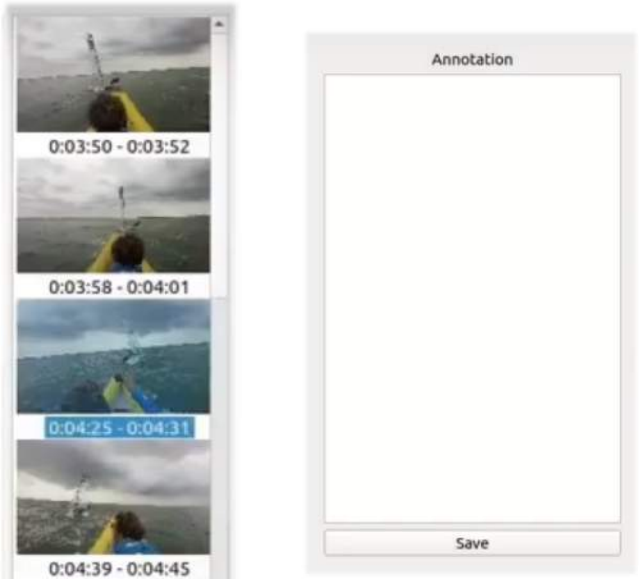


FIGURE 4.25: Timeline as used in the Visual Analysis Framework

List of Intervals The list of intervals provides an overview of all the intervals that are associated with a video, see Figure 4.26a. They allow the user to select an interval based on the thumbnail. The intervals in the list are linked with the intervals in the timeline, clicking them will place the cursor of the timeline at the correct location in time. Intervals in the list can be marked as important, deleted or annotated by the user. Intervals can be annotated by adding the comments using the Annotation component of the framework, as can be seen in Figure 4.26b.



(A) Example of a list of manoeuvres with thumbnails and interval timings.

(B) Window allowing the user to annotate the intervals in the list.

FIGURE 4.26: List of intervals and annotation components of the Visual Analysis Framework.

Complete Visual Analysis Framework When we assemble these components together in a singe User Interface we end up with the framework such as the one that can be seen in Figure 4.27. For an enlarged version see Figure A.6 in Appendix A. This is the complete visual analysis framework, we will evaluate it with the intended users in Section 5.4

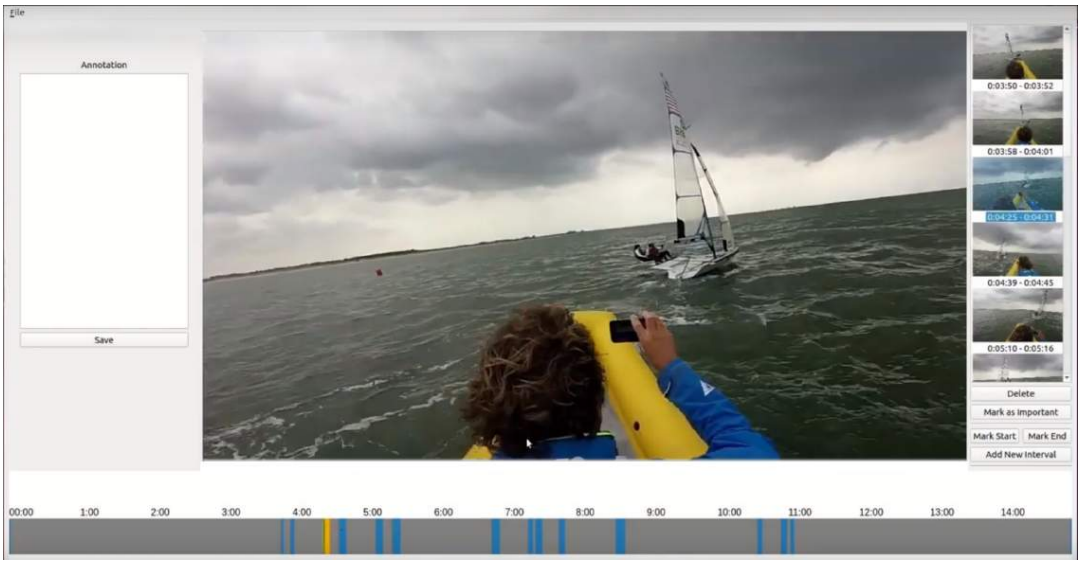


FIGURE 4.27: The complete Visual Analysis Framework with all components.

4.6.2 Implementation Details

The Visual Analysis Framework was implemented using a combination of libraries and programming languages and was tested on the Ubuntu Linux Platform. Most of the interface was built using Python and Qt, using Python bindings provided by Pyside2 [3]. All of the features were implemented using Qt except for the Timeline and the Mediaplayer playing the videos.

The Timeline was implemented using JavaScript and D3[4]. The Javascript code runs inside a QWebEngineView which communicates events with the other components of the interface via a QWebChannel. This way, we can communicate and synchronize click events between all components making it a connected interface.

To have a robust and versatile mediaplayer we make use of Video Lan Client Mediaplayer[5], better known as VLC. Using Qt and the Python bindings for VLC[6] we make calls to the VLC API. This requires VLC to be installed on the platform. The Timeline and VLC calls are synchronised to allow for the playback at the location a user selects in the timeline.

[3] `https://pypi.org/project/PySide2/`
[4] `https://d3js.org/`
[5] `https://www.videolan.org/`
[6] `https://pypi.org/project/python-vlc/`

Chapter 5

Evaluation & Results

In this chapter we will evaluate the proposed methods (see Chapter 4). First, we will evaluate the stabilization quality quantitatively using quality metrics and qualitatively using a small user study with seven coaches. We evaluate whether the stabilized videos are easier to analyze as a coach. Second, we evaluate the Detection and Tracking accuracy with two experiments using available relevant test videos. In these experiments we compare a manually constructed ground truth with the output of the proposed methods in Sections 4.3 and 4.4. Third, we evaluate the Manoeuvre Detection by comparing the output of the proposed method with a ground truth and determine the sensitivity and accuracy for a small set of relevant test videos. Fourth, we evaluate the proposed Visual Analysis Framework using a user study with seven coaches.

5.1 Stabilization

The stabilization method as described in Chapter 4 was evaluated using videos provided by annalisa, SIC and coaches at NTCZ. Most of these videos were taken using a camera mounted to a pole on the RIB and mostly contain footage of the RIB following a sailing boat around in different sea states and weather conditions. The implemented system takes around 0.134 seconds to process a video frame (1920x1080 pixels, H.264) on a laptop with an i7-4720HQ processor and 16GB of RAM. This is the single thread performance where the computation bottleneck is applying the median blur and calculating the probabilistic Hough transform for line detection, both methods take on average between 60 and 70 ms each per video frame. A speed up could be achieved by a parallelized processing of the video frames or by cropping the video frames around the detected horizon, see Section 4.2.6. An experiment with cropping the video frames around the horizon to between 40-50 percent of the original size resulted in a speedup of around 2. Experimental results of some of the videos used, without applying the aforementioned methods to achieve a speedup, can be found in Table 5.1 below.

To be able to observe the difference it is advised to view the original and stabilized versions of the videos[1]. These videos are available for comparison and convey the difference in viewing experience a lot better than an image sequence. Two examples of such image sequences, comparing the original and stabilized video, can be found in Figures 5.1 and 5.2.

The difference between the original and the stabilized frames from the video is clearly visible in Figure 5.1. The waves caused the RIB to roll and therefore rotated the camera with respect to the horizon. This rotation and translation is compensated by the stabilization to end up with the bottom row of images in Figure 5.1.

[1] `http://tiny.cc/Stable`

TABLE 5.1: Video stabilization experimental results.

Description	Length (s)	Bitrate (kbps)	Processing time	FPS	Total Time*	FPS*
Garmin VIRB @ (1920x1080, H264)	57	24965	03:49.03	7.47	04:40.18	6.11
Garmin VIRB @ (1920x1080, H264)	14	24966	00:58.87	7.44	01:11.72	6.11
Garmin VIRB @ (432x240, H264)	10	62208	00:02.37	105.83	00:02.96	84.80
iPhone @ (1920x1080, H265)	8	8166	00:35.46	6.40	00:43.37	5.23
Amcrest @ (1280x720, H264)	332	1771	03:03.02	27.25	04:04.63	20.38
Amcrest @ (1280x720, H264)	286	4706	04:21.12	16.44	05:22.60	13.30
Amcrest @ (1280x720, H264)	170	2388	01:49.57	23.40	02:21.72	18.00

* Total includes the encoding and writing of the video frames to disk.

FIGURE 5.1: Comparison of Original (top row) and Stabilized frames (bottom row) (432x240 pixels source)

FIGURE 5.2: Comparison of Original (top row) and Stabilized frames (bottom row) (1920x1080 pixels)

When the water surface is calmer, such as in Figure 5.2, the rotations and translations needed to stabilize the original frames are not as extreme. The geometric distortion caused by the wide-angle lens of the camera, in this particular case a Garmin VIRB, is however still visible in the stabilized frames. When these distortions are more extreme they can reduce the quality of the stabilization method or cause it to fail. Examples of the limitations, mentioned in Section 4.2.5, will be discussed in Section 5.1.3 below.

5.1.1 Stabilization Quality

To quantify the difference in quality between the stabilized video and the original we make use of the quality metric Peak Signal-to-Noise ratio (PSNR). We use the PSNR because it is a simple and understandable metric. Despite a lot of criticism on this metric it is a good metric for comparative quality assessment when keeping the video content the same according to Korhonen and You [56].

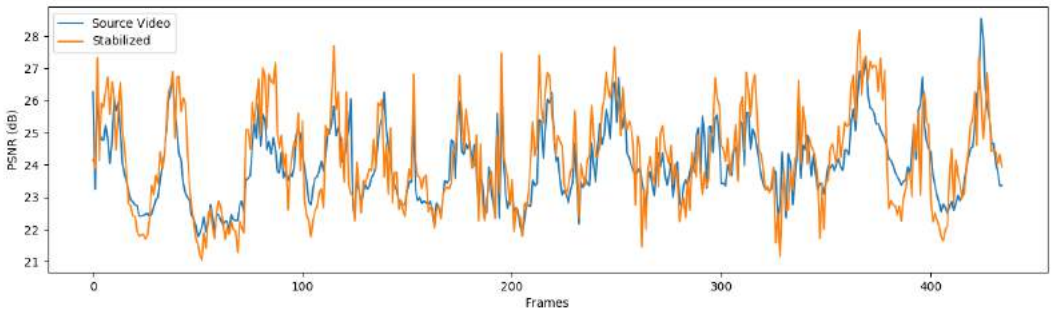
The Peak Signal-to-Noise ratio between consecutive frames in dB is defined as

$$PSNR = 10 \cdot log_{10} \left(\frac{MAX_I^2}{MSE} \right) \tag{5.1}$$

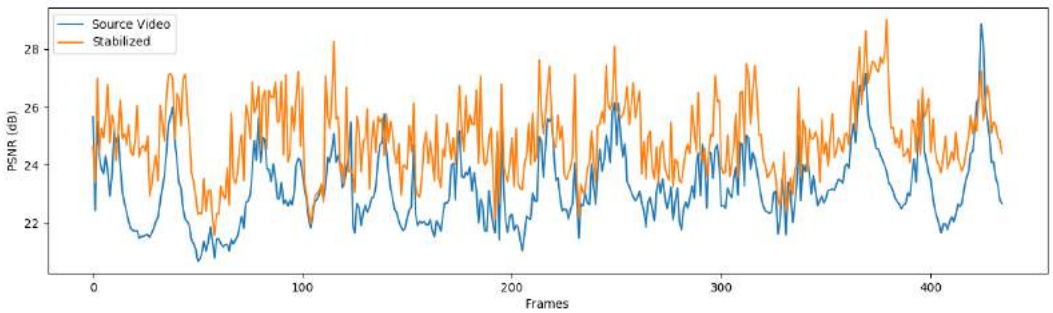
Where MAX_I is the maximum possible pixel value of image I and the Mean-Squared-Error MSE for consecutive frames, with dimensions (M, N) is defined as:

$$MSE(n) = \frac{1}{MN} \sum_{j=0}^{M} \sum_{i=0}^{N} [I_n(i,j) - I_{n+1}(i,j)]^2 \tag{5.2}$$

This relatively simple quality metric gives an idea of the improvement in quality, if any. Next to this, we will also use it to compare the cropped stabilized versions to the non-cropped stabilized versions. In Figure 5.3a a graph of the PSNR can be found of a video and its stabilized counterpart.



(A) PSNR graph for Source and stabilized video.



(B) PSNR graph for Source and stabilized video with cropped frame.

FIGURE 5.3: Graphs of non-cropped and cropped PSNR values.

From Figure 5.3a we can observe that the stabilized version is only slightly better than the original according to this quality metric. This small difference is for the most part caused by the moving edges due to the applied transformations and the black background filler. When we look at the cropped version of the video, see Figure 5.3b, we observe that the PSNR is higher for the stabilized video. An example of

the difference between the original video and the cropped version can be found in Figure 5.4.

FIGURE 5.4: Original frame and orange edge signifying what the frame would be cropped to.

To summarize the PSNR graphs into single values we use the Interframe Transformation Fidelity (ITF), which is the PSNR between consecutive frames averaged over the whole video. Experimental results for 4 videos taken under different conditions can be found in Tables 5.2 and 5.3. From these results we can conclude that for these experiments the stabilized cropped version is always better than the non-stabilized cropped version. This suggests that the stabilization does improve the quality of the videos. However, according to the quality metric, not cropping the edges reduces the quality. Though, as stated before in Section 4.2.4, cropping can cause problems as well. We leave the problem of removing the black areas by cropping or by inpainting as future work.

What is more important though, is if the coaches that have to work with the videos think the stabilized videos are actually better. This will be evaluated next.

Conditions	**ITF Source (dB)**	**ITF Stabilized (dB)**
Sunny & Calm	27.20	27.44
Large Waves & Rain	30.34	26.12
Medium Waves & Cloudy	23.99	24.21
Large Waves & Cloudy	30.83	29.93

TABLE 5.2: ITF for source and stabilized videos taken under different conditions.

Conditions	**ITF Cropped Source (dB)**	**ITF Cropped Stabilized (dB)**
Sunny & Calm	26.58	28.00
Large Waves & Rain	30.05	30.62
Medium Waves & Cloudy	23.23	24.88
Large Waves & Cloudy	30.78	30.98

TABLE 5.3: ITF for Cropped source and stabilized videos taken under different conditions.

5.1.2 User study

A user study with seven coaches was conducted to evaluate the stabilization method (see Section 4.2). For a list of the participants see Table 5.4. The goal of this evaluation was to determine if the users (the coaches) prefer the stabilized videos or the original videos and what the motivations are behind their preference.

To evaluate their preference the users were shown three pairs of videos, each original video of the pair taken under different conditions. One video in sunny weather and with relatively calm sea state, one cloudy and medium waves and the last one cloudy and large waves. Each pair of videos consisted of the original video and a stabilized version , for an example see Figure 5.5. Then, for each pair the following question was asked:

> *Which video (1 or A) is easier to analyze? (1) - Strong preference for Video 1 ; (3) - Both videos are equally easy/difficult); (5) - Strong preference for Video A*

The user would then be presented with a Likert scale [57] and had to select one of the 5 options (1-5). All the votes and motivations were collected of the seven coaches and will be discussed in **Results** below.

FIGURE 5.5: A pair of videos used in user study. Video 1 being the original video and Video A the stabilized version.

User study participants

TABLE 5.4: List of coaches who participated in the user study.

Results

From the summarized results in Figure 5.6 we can conclude that there is a preference for the stabilized versions of the videos, as 10 of the 21 votes either prefer or strongly prefer the stabilized version of the videos. The results per video can be found in Appendix A, Figures A.3, A.4 and A.5. Motivations given for these choices in favor of the stabilized versions of the videos is that it makes looking at the details easier and that

"the movements of the video are caused by the RIB, which are totally irrelevant". Moreover, one coach states that it is easier to focus on the boat in the stabilized videos.

Two of the coaches mention that the videos are equal for them, both the original and stabilized version were "okay".

However, not all coaches agreed with the stabilized version making it easier to analyze. One of the coaches had a slight preference for the stabilized version but was concerned what would happen in rough conditions. Another coach, who had a strong preference for the original video, stated the following: "Stabilized video would be better if surroundings are cropped; really distracting at the moment to have the changing sides; cause those are continuous references for the brain".

We can conclude that the stabilized version is considered to be the better version, but in future work the distracting moving edges should be addressed.

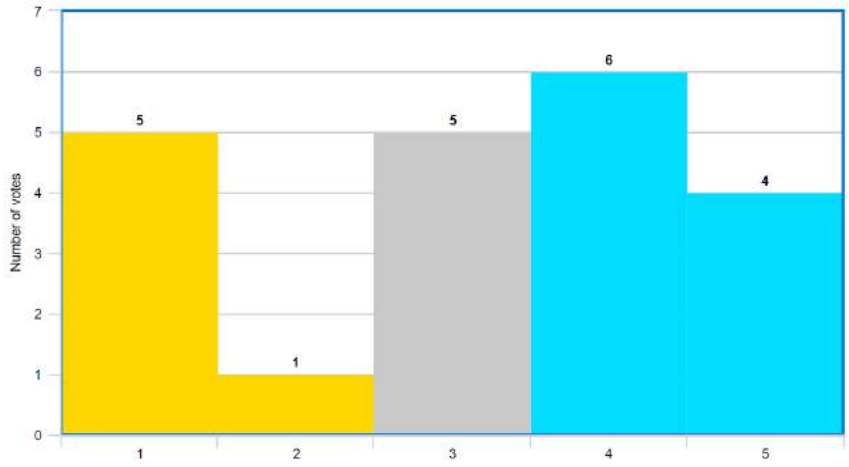


FIGURE 5.6: Results of the votes in the stabilization user study. Votes for 1 are strongly preferring the original video, votes for 3 are both videos are equal and votes for 5 are strongly preferring the stabilized video.

5.1.3 Examples of limitations

One of the limitations of the stabilization method comes from the assumption that the horizon is more or less a straight line, as discussed in Section 4.2.5. An example of a failure case because of the geometric distortion can be found in Figure 5.7a. It is clear that the horizon line in this frame is not in fact a straight line and therefore causes the method to fail at this point.

As mentioned in Section 4.2.5, raindrops on the lens of the camera can distort the image too much resulting in the stabilization method not working properly. For an example with such conditions see Figure 5.7b.

(A) Example video frame with geometric distortion.

(B) Example video frame with rain drops.

FIGURE 5.7: Examples of situations limiting the performance of the stabilization method.

5.2 Detection and Tracking

To evaluate the Detection and Tracking two experiments were conducted. The purpose of these experiments is to evaluate the detection and tracking methods, as described in Sections 4.3 and 4.4. In the first experiment we compare the detection and tracking performance of a boat in a video with a ground truth. Then, in the second experiment we will compare the combined detection and tracking performance for the boat and person with a ground truth. The experiments and results are discussed in more detail in the following sections.

5.2.1 Boat Detection

First, a ground truth was constructed for a non-stabilized video. The video sequence used in this experiment was of a manoeuvre scenario where we closely follow the boat and the boat and person could always be detected. Constructing the ground truth was done by manually annotating the video with labeled bounding boxes. For this experiment only the boat was our object of interest, the sailors on the boat are not considered. Next, the implementation of the methods for detection and tracking (see Sections 4.3 and 4.4) was used to process the same video. The output, a bounding box around the boat, was then compared to the ground truth by using the Intersection over Union (IoU). This measure can be calculated per frame with equation 5.3, where G_i is the ground truth bounding box of the object and D_i is the detected/tracked bounding box.

$$IoU = \frac{|G_i \cap D_i|}{|G_i \cup D_i|} \tag{5.3}$$

With this measure we measure the overlap between the ground truth and the object detected using the algorithms. In this experiment we measured the Intersection over Union for the detection and the combined detection and tracking. The results for the video in this experiment can be seen in Figure 5.8. From these results we would conclude that using the detection only is better, as the IoU or "similarity" is higher than

the combined Detection & Tracking. However, this difference in IoU is mostly because the bounding box for the combined Detection & Tracking is almost always a larger bounding box than the ground truth, where using the Detection only is usually smaller. Although for accurate detection you would normally want to be as close as possible to the ground truth this does not matter as much for our method. As explained in Section 4.5, we will use the x-coordinate of the middle of the bounding box in the video frames. Therefore, when we compare the Euclidean Distance for these one dimensional points, we see that the Detection-only does not always outperform the combined Detection & Tracking (see Figure 5.9). This difference is caused, as mentioned before, by the difference in bounding boxes (see Figure 5.10). We can observe from Figure 5.10a that the middle of the bounding box (in x direction, horizontal axis) is more off than the bounding box in Figure 5.10b, which was created using combined Detection and Tracking.

An IoU of 0.6 for the combined Detection & Tracking (see Figure 5.8) is not very accurate. Even though the middle of the bounding box created by the combined Detection & Tracking is more accurate than the IoU of 0.6 suggests, the Detection only is more accurate. Therefore, when possible the Detection bounding box should be used. However, the Detection method does not always produce an output which was the reason to include Tracking as mentioned in sections 4.3 and 4.4. More on this in section 5.2.3. In future work the Detection method could be improved upon to try to track the boat and persons by using Tracking by Detection. This would mean that the boat and person need to be detected in every frame, which is not the case right now.

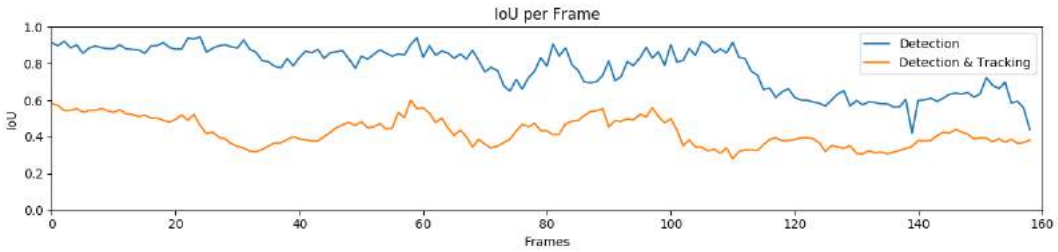


FIGURE 5.8: Intersection over Union per frame of a test video for Detection and combined Detection & Tracking, compared with a manually constructed ground truth.

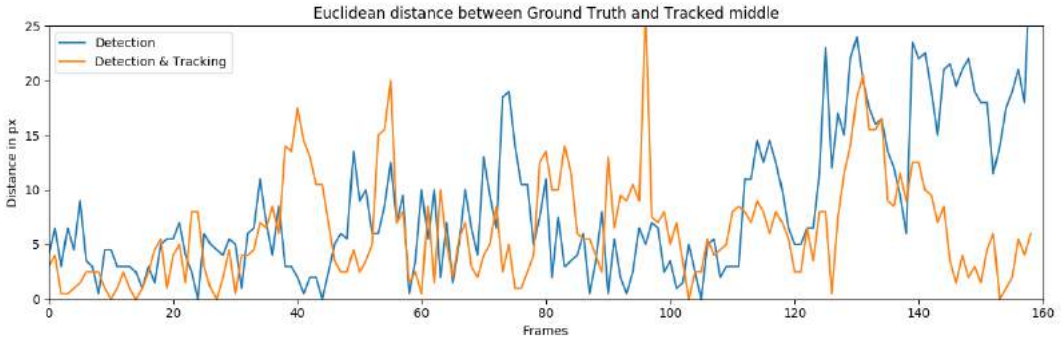
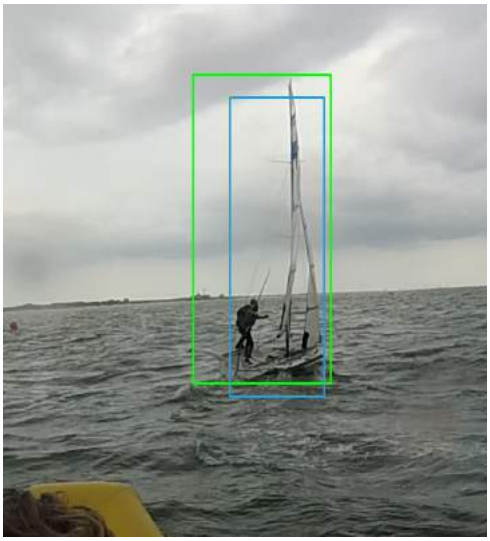
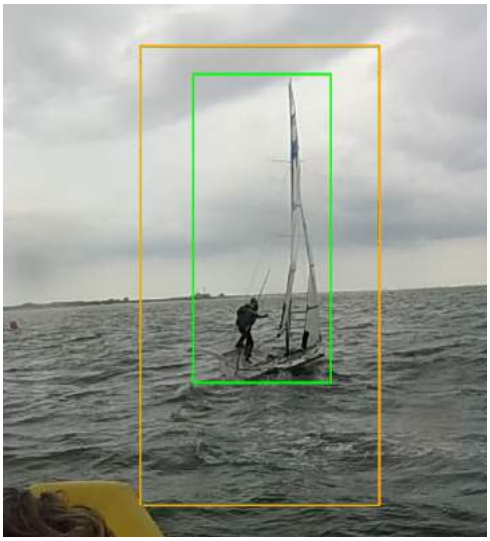


FIGURE 5.9: Comparison of Detection-Only and Detection & Tracking regarding the Euclidean distance to the middle of the Ground truth Bounding Box from the method output Bounding Box. Lower is better.



(A) Bounding box around boat generated by Detection only (Blue) compared with ground truth bounding box (Green).



(B) Bounding box around boat generated by Detection and Tracking (Orange) compared with ground truth bounding box (Green)

FIGURE 5.10: Bounding boxes for Detection only, and combined Detection and Tracking. Compared with a ground truth bounding box around the boat (Green).

Stabilized vs. Non-Stabilized As an extension of the boat detection experiment we compared the accuracy for Stabilized and Non-Stabilized videos. As stated before in Sections 4.1 and 4.2, we assume that the stabilized videos are beneficial to the performance of the Detection and Tracking methods. To test this, similar to the boat detection experiment, a ground truth was created. However, for this extension we created a ground truth for the original video and the stabilized version of this video. The Detection and Tracking performance was then measured using the Intersection over Union, the results for both videos can be found in Figure 5.11. For the Detection and Tracking the average IoU improved from 0.43 for the original to 0.60 for the stabilized video. From these results we can conclude that for this video our assumption holds. There

is a significant difference in performance for the combined Detection and Tracking. When we look at the performance using only the Detection method we do not see a significant difference. This is because the Tracking method does use the time-coherent information of a sequence of frames whereas the Detection method just uses separate frames. Therefore the tracking benefits from the reduced motion of the object of interest.

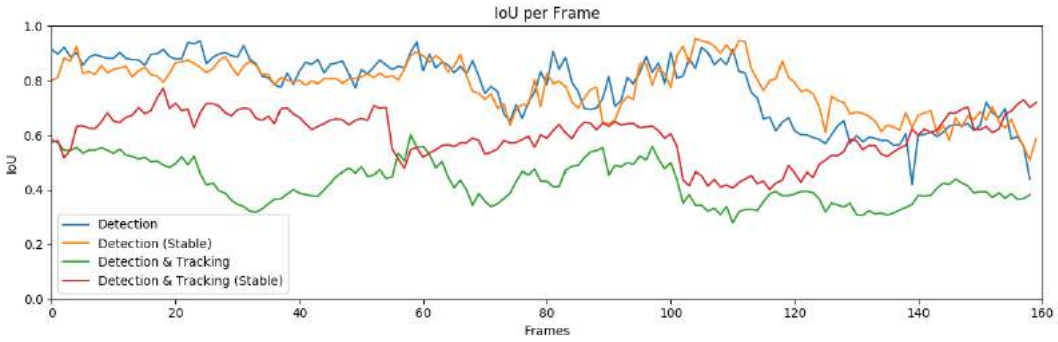


FIGURE 5.11: Comparison of the IoU per frame for a stabilized video and the original video.

5.2.2 Boat and Helmsman

In this experiment both the boat and helmsman (the sailor in control of the rudder) were objects of interest. Similar to Experiment 1, a ground truth was constructed and used to compare against. For the comparison the same measure, as described with Equation 5.3, was used. The Intersection over Union was calculated for the boat and helmsman separately, the results can be seen in Figure 5.12. The video contained footage of the crew of a 49er doing a Tack manoeuvre. Examples of the detected bounding boxes compared with the constructed ground truth can be seen in Figures 5.13a and 5.13b. For the first fifteen frames the person was not detected, resulting in a IoU of zero (see Figure 5.12). Next to this, the detection jumped to the other sailor for a brief period from frame 19 until 30 which resulted in a IoU value of zero as well. However, this does not cause problems because these jumps and gaps will be accounted for in the next step of the pipeline, the Manoeuvre Detection method (see Section 4.5). Again, with an average IoU of around 0.6 is not really accurate. Therefore the Detection only should be used when available, but whenever the boat or persons are not detected we have to rely on the Tracking method.

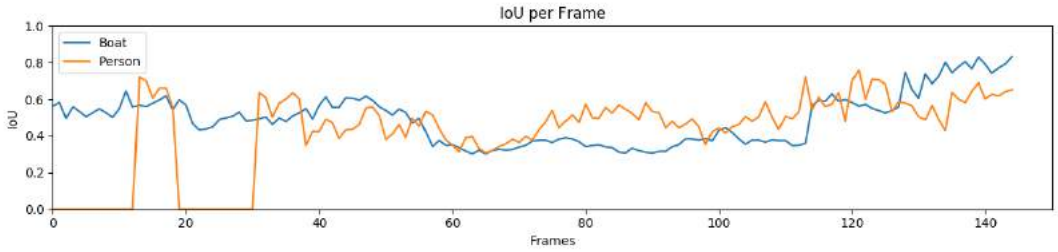
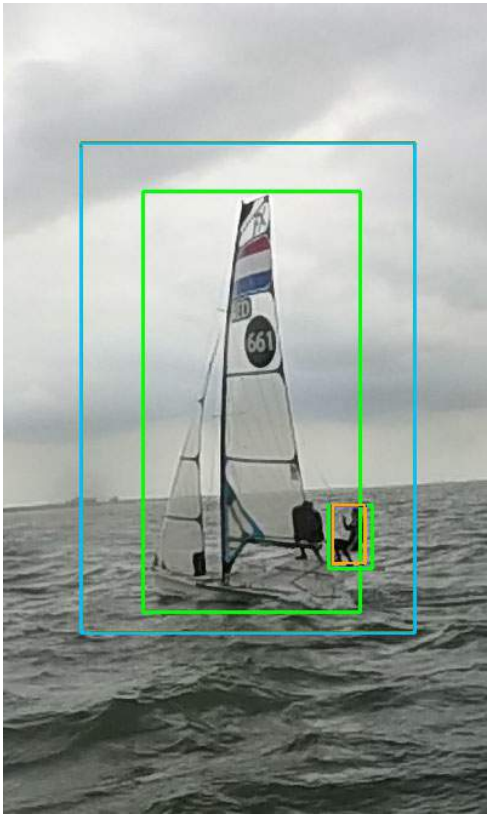
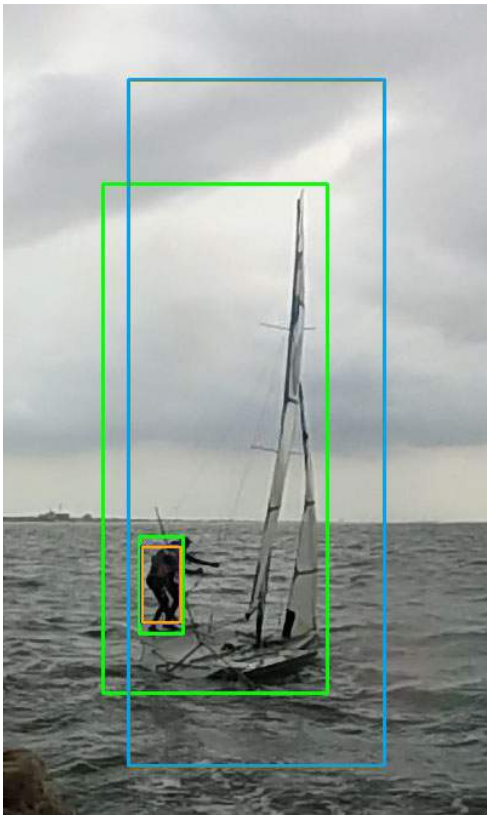


FIGURE 5.12: Intersection over Union per frame of a test video with combined Detection & Tracking, compared with a manually constructed ground truth. Objects of interest are the boat and the helmsman.



(A) Before Tack. Bounding box around boat (Blue) and helmsman (Orange) generated by Detection and Tracking compared with ground truth bounding box (Green)



(B) After Tack. Bounding box around boat (Blue) and helmsman (Orange) generated by Detection and Tracking compared with ground truth bounding box (Green)

FIGURE 5.13: Bounding boxes generated using Detection and Tracking. Compared with a ground truth bounding box (Green).

5.2.3 Combined Detection and Tracking

Although from Experiments 1 and 2 above it may seem that using only detection results in a higher accuracy, this is only partly true. While it is true that the bounding box generated using only Detection often has a higher Intersection over Union value, it sometimes results in too

small bounding boxes, which can cause problems in next steps (see Figure 5.10a). For example because the sailors could be cropped out when using this bounding box. Of course, we could add a margin and still use this. However, what the experiments do not show is that often the Detection method fails to detect the persons in the video. Additionally, from time to time the detection network struggled with the detection of the boat as well. For an example of the advantage of the combined detection and tracking see Figure 5.14. Note that this is a different video sequence than in the Experiments and is presented here to illustrate the point that we can not only rely on the Detection method. In this graph we see the Detection only (DNN Detect) and the combined Detection and Tracking (Tracking). From this graph we can conclude that with the combined Detection and Tracking we are able to have a much more continuous stream of information, which in turn should result in a more robust Manoeuvre Detection. The manoeuvre will be evaluated next in Section 5.3.

The previously mentioned empirical evidence in this evaluation confirm our decision to use tracking together with detection.

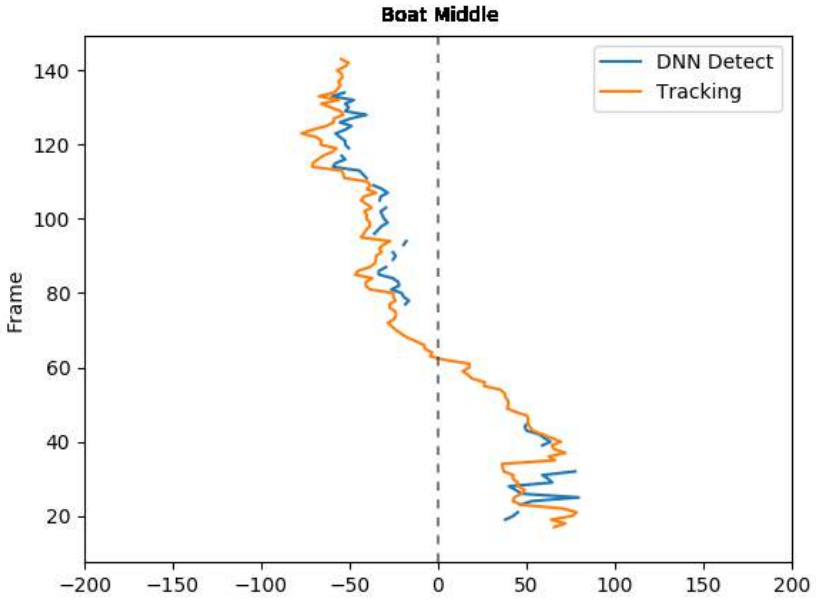


FIGURE 5.14: Comparison of person location with respect to the middle of the boat using Detection only (DNN Detect) and combined Detection and Tracking (Tracking). The x-axis is distance in pixels from the middle of the boat.

5.3 Manoeuvre Detection

The performance of the manoeuvre detection was evaluated using three different relevant video sequences, recorded by a camera mounted to a RIB. We evaluate the sensitivity of the manoeuvre detection method as well as the accuracy of the zero crossing detection method, both methods are described in Section 4.5. To evaluate the sensitivity we compare the output of the Manoeuvre Detection method with a manually constructed ground truth. The output of the Adapted Edge Focusing algorithm will

be compared with a manually constructed ground truth as well. This will be discussed in more detail below.

5.3.1 Sensitivity

To evaluate the performance of the manoeuvre detection we compare the output of the manoeuvre detection with a manually constructed ground truth. This ground truth consists of the manually annotated manoeuvre intervals that can be seen in the videos. We will compare this with our manoeuvre detection method and quantify the True Positives, False Negatives and False Positives. The amount of videos available, of a RIB with the camera mounted following a 49er boat, is too low to draw strong conclusion about the performance of the manoeuvre detection. However, it should give an idea of what can be expected when conforming to the requirements and assumptions, as stated in Section 1.1.2.

The videos used for the comparison contained 14 manoeuvres that were manually annotated as the ground truth. The results of the comparison can be seen in Table 5.5.

	Ground Truth	TP	FN	FP	Sensitivity	Corrected FP	Corrected Sensitivity
Video 1	3	1	2	12	∼33%	2	∼66%
Video 2	7	4	3	7	∼57%	0	∼80%
Video 3	4	3	1	3	∼75%	2	∼75%
Total	14	8	6	22	57.14%	4	72.72%

TABLE 5.5: Results from the comparison between ground truth and output of Manoeuvre Detection. True Positives (TP), False Negatives (FN) and False Positives (FP) and the calculated sensitivity. The situations that violate the requirements and assumptions are discarded and the remaining manoeuvres are referred to as the "Corrected".

When we remove the situations where the assumptions do not hold from the set of ground truth manoeuvres we see that the average sensitivity of the method is 72.72% for these videos (See Table 5.5). The amount of data available is not large enough to draw strong conclusions on the sensitivity, but does give an indication of what can be expected when adhering to the requirements and assumptions. The violated requirements and assumptions are in most cases caused by the RIB not following the sailing boat or the sailing boat being too far away to detect and track the persons on it. For an example of both these situations see Figure 5.15. When we look at the results in Table 5.5 without removing the situations where the assumptions do not hold we see that the sensitivity of the Manoeuvre Detection method is 57.14%. This relatively low sensitivity is partly because of the violation of the requirements and assumptions.

(A) Sailing boat disappearing off screen while RIB continues straight ahead, limiting the ability to detect the manoeuvre.

(B) Sailing boat too far away to detect the persons on the boat, limiting the ability to detect the manoeuvre.

FIGURE 5.15: Limitations for manoeuvre detection caused by violated assumptions.

The number of False Positives from the three videos combined was on the high side with 22, as can be seen in Table 5.5. However, the vast majority of these False Positives was caused by the RIB lying stationary next to the sailing boat for some time. This limitation, as described in Section 4.5.3, comes from the violated assumption that the RIB is following the sailing boat around from behind. For an example of the sailing boat lying stationary next to the RIB see Figure 5.16. When we discard all the situations where a False Positive was generated by one of the assumptions being violated, we end up with a corrected False Positive count of four. Most of the violated assumptions (13 out of 22) were the sailing boat lying next to the RIB. Other violated assumptions causing False Positives were the RIB not following the sailing boat or other boats crossing between RIB and the followed sailing boat.

FIGURE 5.16: Example of situation causing False positives, sailing boat next to the RIB violates assumption and in turn generates false positives for the Manoeuvre Detection.

5.3.2 Zero-Crossing Accuracy

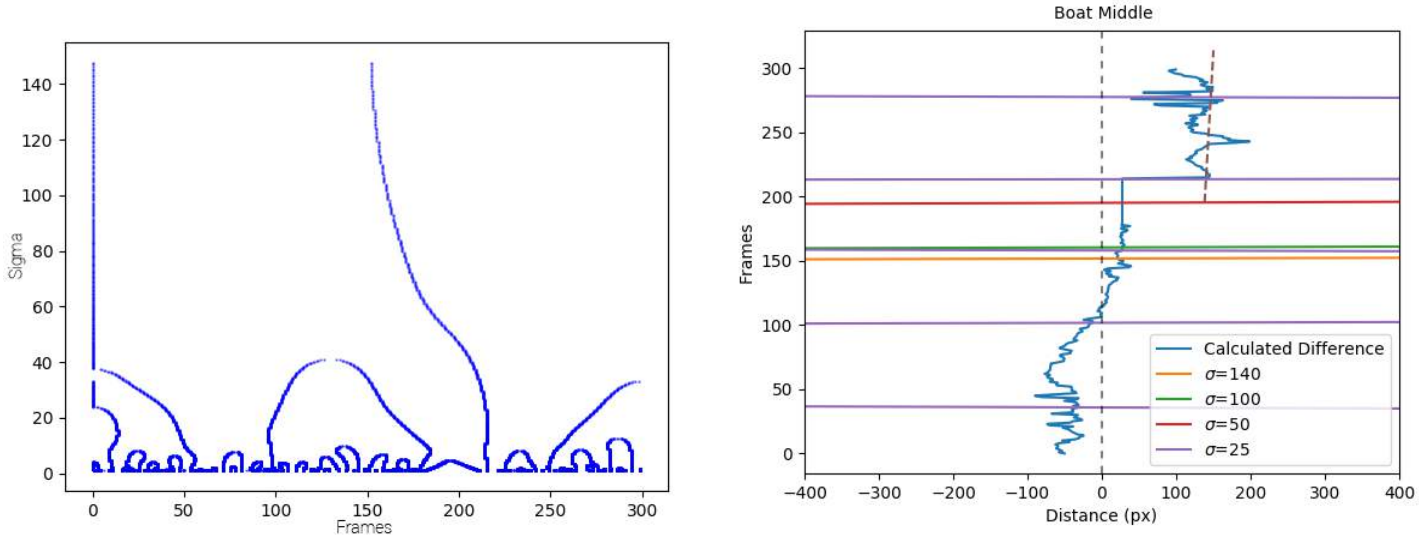
To evaluate the Zero-Crossing accuracy we created a set of zero-crossings by applying the Adapted Edge Focusing method (see Section 4.5) on a

set of videos. These where then compared with a manually constructed ground truth to evaluate the accuracy in terms of frames. A 100% accurate zero-crossing would be the method outputting the same video frame as the ground truth as the point in time where the zero-crossing occurred. For the results of this comparison see Table 5.6. From these results we can conclude that the Adapted Edge Focusing is relatively accurate. For 20 cases the method was accurate to the exact frame, for the remaining 7 of the set this was not the case. With a median offset of 26 frames and an average of 45 frames this means that in practice, with a frame-rate of 30 frames per second, we would be off by about a second.

Total	Accurate	Inaccurate	Avg. Offset	Med. Offset
27	20	7	$\sim$45 Frames	26 Frames

TABLE 5.6: Results for the accuracy of the Zero-Crossings.

Although the precise frame of the zero-crossing is not that important we encountered one case where the Adapted Edge Focusing was off by 100 frames. For an example of the limitations of the Adapted Edge Focusing, as described in Section 4.5.3, see Figures 5.17a and 5.17b. From these figures we can see that coarse to fine search starts around frame 151 for a LoG filter with a σ of 140. However, instead of tracking the edge to frame 115, which is where the actual zero-crossing occurs, the edge is tracked to frame 215. This is caused by the missing data and in this case causes the Edge Focusing to generate the output with an offset of 100 frames. With a recording at 30 frames per second this would mean that we are more than three seconds off with detecting the middle of the manoeuvre. In most cases this should not be a problem, as searching for the stable lines should still find the start and end if the start and end can be found in the window around the middle of the manoeuvre.



(A) (B)

FIGURE 5.17: Example of missing data causing inaccuracy in the output of the Adapted Edge Focusing method. (A) Graph of "signatures" using LoG at different values for σ, using the Calculated Difference in 5.17b as input data. (B) Graph of calculated difference data filtered using the LoG at different values for σ, used in the Adapted Edge Focusing. Missing data causes the sudden jump in value around frame 215, causing the coarse to fine tracking output the wrong frame as zero-crossing point.

5.4 Visual Analysis Framework

To evaluate the created Visual Analysis Framework a user study was conducted. Seven coaches participated in the user study, a list of the participating coaches can be found in Section 5.1.2. The goal of this user study is to determine if the coaches think that this would be a useful addition for their coaching or not. A working prototype, which can be seen in Figure 4.27 in Section 4.6, was shown to the participating coaches and evaluated using a questionnaire. This questionnaire consisted of the following questions below:

Question	Answer Type
What aspects of this framework, if any, would be useful for coaching?	Open Question
What features are you missing in this framework?	Open Question
What features are not useful?	Open Question
How likely is it that you would use this in coaching?	Likert Scale
How useful, in your opinion, is the timeline with marked intervals/manoeuvres in the framework?	1-10 Scale

5.4.1 Results

The summarised results of the questions and interviews can be found below. The results are summarised per question in the paragraphs below,

the full list of answers can be found in Appendix B Figures B.1 and B.2.

What aspects of this framework, if any, would be useful for coaching? Three of the coaches mentioned that all of the aspects could be useful in coaching. Others named specifically the ability to mark manoeuvres and making notes that are attached to these intervals. One coach did not think any of the aspects of the framework would be useful in coaching. The motivation for this was that there was no detail to be seen in the videos, which suggests that this coach was evaluating the quality of the videos rather than the framework around the videos.

What features are you missing in this framework? Two of the coaches, one in the questionnaire and another in an interview, stated that they were missing an easy way to share and store the clips. The prototype did not contain functionality for sharing and storing the clips and notes. Another feature that was mentioned was the ability to draw on the videos, as well as labeling/naming the clips.

One coach mentioned the ability to zoom in on the boat to be able to see more detail, because for the relatively fast sailing boats it is difficult to stay close. Then, if the quality of the video is high enough, you could automatically zoom in and follow the boat. This was a particularly interesting suggestions as we have the tracking data available already. Therefore, if the video resolution is high enough this could be an interesting addition to the framework.

What features are not useful? Besides the notes that the videos are too far away, which is not necessarily a feature of the framework, the most interesting comments were on the list of manoeuvres. Although we assumed that the thumbnails would give an indication of the contents of the interval, this is not the case. One coach stated that everything looks the same in the thumbnails and therefore it is not very useful. This in line with the comments of another coach on the previous question (*What features are you missing in this framework?*), who stated that you need a way to label/name the intervals to be able to distinguish between them. This is an important aspect that has to be taken into account in the next iteration of the framework design.

How likely is it that you would use this in coaching? The results for this question can be found in Figure 5.18 below. Two of the coaches voted low on the Likert scale (1 and 2). On average we can conclude that the coaches would likely use this in coaching. The coach which voted the lowest was the coach who stated that no detail could be seen in the videos, and therefore was not useful. The other coach who voted low did find the aspects of the framework useful and found the timeline very useful. However, this coach found it not likely that he would use this himself in coaching, but could see the potential value in a framework like the one presented.

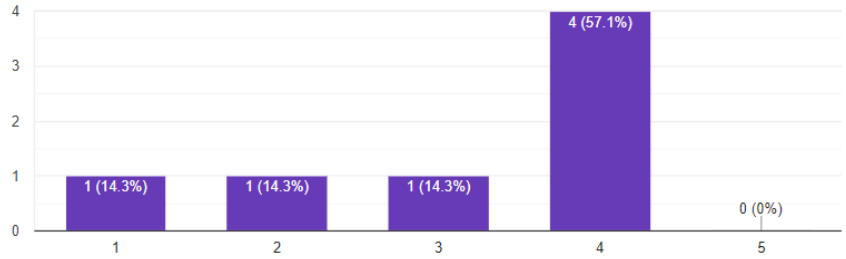


FIGURE 5.18: Results of the answers of the question: "How likely is it that you would use this in coaching?". Answers were in the form of a Likert scale.

How useful, in your opinion, is the timeline with marked intervals/manoeuvres in the framework? This question was asked to evaluate whether the timeline makes sense to the coaches and if they think it is a useful addition. We assumed that this would be a good method to show the intervals in time for the video. The results in Figure 5.19 below indicate that this is indeed the case. With an average of 7.43, and no votes below a value of 6, we can conclude that all coaches consider the timeline to be a useful aspect of the framework.

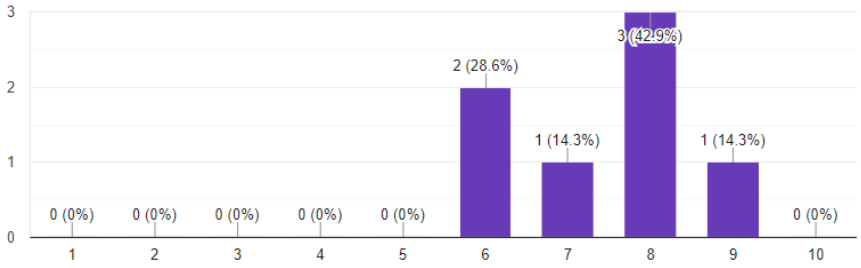


FIGURE 5.19: Results of the answers of the question: "How useful, in your opinion, is the timeline with marked intervals/manoeuvres in the framework?". Answered on a discrete scale from 1-10.

Chapter 6

Conclusions and Future Work

Switching to the new situation with cameras mounted to a pole created some problems and at the same time provided opportunities. In this thesis we presented a pipeline that addresses the problems of the videos regarding stability and leverages the available video data using multiple steps. The goal of this research was to improve the visual analysis of recorded training sessions used to train sailing athletes in the dinghy class. We did so by stabilizing the videos using the horizon as a visual cue, making the movements and details of the sailors easier to analyze. Next to this, we extract manoeuvres from the recorded training sessions by tracking the location of the boat and sailors in the videos. These extracted intervals are then visualized in the Visual Analysis Framework to be able locate these in the recorded videos, without having to search through the entire recorded session by hand. These contributions combined improve the visual analysis of the recorded training sessions while using only the video data and minimal user input.

To deal with the unstable videos, due to the camera mounted to the RIB, we created a stabilization method based on the horizon line. Under most circumstances this method is able to compensate the induced motion. The majority of the users in the user study agreed to the stabilized videos being easier to analyze. The moving borders caused by the transformations of the stabilized video are, however, too distracting according to some users. To improve the stabilized videos this would have to be addressed. Next to the improved stability of the videos, making them easier to analyze, the stabilization has a positive effect on the accuracy of the tracking method. During experiments (see Section 5.2.1) the average Intersection over Union went from 0.43 to 0.60 for the stabilized video using the proposed Detection and Tracking method.

The Detection and Tracking is used to provide input data for the manoeuvre detection. The combination of the Detection and Tracking provide data of reasonable quality. Although the IoU with a manually constructed ground truth is usually around 0.6, this is most of the time accurate enough to be able to detect the manoeuvres. In most videos the Detection and Tracking of the boat was a lot more consistent than tracking the persons on the boat. In most cases this is caused by the distance between the camera and the boat, making the persons too small to detect using this version of the Deep Learning network. The Tracking method does however make sure that, in most cases, we are able to fill in the gaps where the Detection does not detect the persons with enough confidence.

The data generated by the Detection and Tracking is used as input for the proposed Manoeuvre Detection method. During experiments with the limited amount of relevant test videos available we found that we reached a sensitivity of around 72%. This was after removing the clearly violated assumptions such as not following the boat around. These assumptions not holding were also the main source of false positives. Although the false positives could easily be discarded by the user, the missed manoeuvres require a different approach in operating the RIB or an improved detection of the persons on the boat. The Zero-Crossing accuracy detection of the Adapted Edge Focusing is, with 20 out of 27 zero-crossings accurate to the frame, reasonably accurate. Most of the inaccuracies were caused by missing data, which would be addressed by an improved Detection and Tracking.

To visualize the intervals and videos a Visual Analysis Framework prototype was created. Using a user study we found that most of the coaches believe they would use the Visual Analysis Framework in coaching. Two of the coaches reported that the list of thumbnails is not useful because in the presented prototype every thumbnail looks the same. They suggest having the option to label or name them to be able to quickly distinguish between the different intervals. Next to this, some of the coaches stated that the observable detail in the videos was too low because the boat was too far ahead of the recording camera. One of the suggested features that came forward during the user study is the option to (automatically) zoom in on the boat. From the user study we can conclude that the proposed timeline is a useful feature of the Visual Analysis Framework. The coaches responded with an average of 7.43 out of 10 and no vote lower than 6 on the question how useful the timeline is, on a discrete scale from 1 to 10.

All in all we were able to address most of our research goals with the presented pipeline. The videos are stabilized under most circumstances, satisfying the ability to see more detail in the videos without the distracting movement. However, the moving edges of the videos make the result not optimal yet. According to most coaches it is, however, an improvement over the original videos. Interesting intervals, in this work only Tacks and Jibes, can be extracted semi-automatically with minimal user input. These are extracted with a reasonable accuracy using the steps in the pipeline, although there is room for improvement. These steps of the pipeline lead to the Visual Analysis Framework presented, which answers the question if we can design a Visual Analysis Framework which improves the analysis of recorded training sessions. The presented pipeline and Visual Analysis Framework is a step forward in the video analysis for sailing coaches, but not yet a perfect solution. There is, however, a number of opportunities which could lead to a more robust and feature-rich Visual Analysis Framework. These will be discussed next.

6.1 Future Work

Although the current pipeline shows potential there are aspects that still need improvement. In the video stabilization step of the pipeline the moving edges caused by applying the transformations should be addressed. This could be done by cropping the video to make sure that all of the moving edges are outside of the cropped. However, this comes with the risk of cropping out potentially important parts of the frame. Another option is to fill up the black background. This could be done by, for example, extending the edges of the frame with a blurred repetition of the edge of the frame. Or, by using inpainting techniques to fill the areas. However, these approaches can still be distracting for the user. This can be less distracting than the current method with the moving edges. The trade-offs between the distracting movement and introduced risks of misinterpretation by trying to address the moving edges should to be researched.

The deep learning network used for detection can be improved upon. For example, by using Transfer Learning to be able to detect the persons on the boat more reliably. Although the network is relatively accurate in detecting persons in all kinds of situations, there is room for improvement for detecting persons on a boat. Especially in the situations where the persons on the boat are further away from the recording camera. This does require a large enough annotated data-set to train the network on, which is currently not available and thus would have to be constructed first for this application. Another approach to improve the detection of the persons on the boat would to add detectable markers or even sensors to the sailors outfits. This does however require the sailors to wear the markers or sensors in a location that is clearly visible while not limiting their ability to operate as they could without these markers. An improved Detection and Tracking method will in turn result in an improved Manoeuvre Detection.

Another approach to improve the Manoeuvre Detection is by training a neural network using videos of manoeuvres. This does however require a good data-set to train this network. This data-set could be constructed using the method proposed in this thesis. By taking the intervals labeled as manoeuvres, after removing the false positives, a data-set could be created and annotated over time and use it to train a new network specifically for detecting manoeuvres.

Although this thesis focused on the video data only, in the future data from other sensors may be available as well. Whenever a wind direction sensor is available, and the assumption holds that the RIB is following the sailing boat, this data can be used to detect the manoeuvres as well. This should, in theory, allow for a much more reliable detection when combined with the proposed method, as from the significant changes in wind angle we can derive when a manoeuvre must have occurred.

One of the future additions for the Visual Analysis Framework could be an automatic zooming feature. As the location of the bounding box around the boat is already available this could be used to implement automatically zooming in around the boat. This would address the concerns of some of the coaches that there is not enough observable detail

in the videos, which is in most cases caused by the sailing boat being too far away. However, this would also amplify the movement and therefore should be investigated if this does indeed improve the observable detail while not limiting the ability to analyze the videos easily.

The presented pipeline already shows potential. By adding the aforementioned features and techniques to the pipeline and Visual Analysis Framework we will create an even better video analysis framework. Which in turn could improve the performance of the sailors and result in more medals and wins.

Appendix A

Additional figures

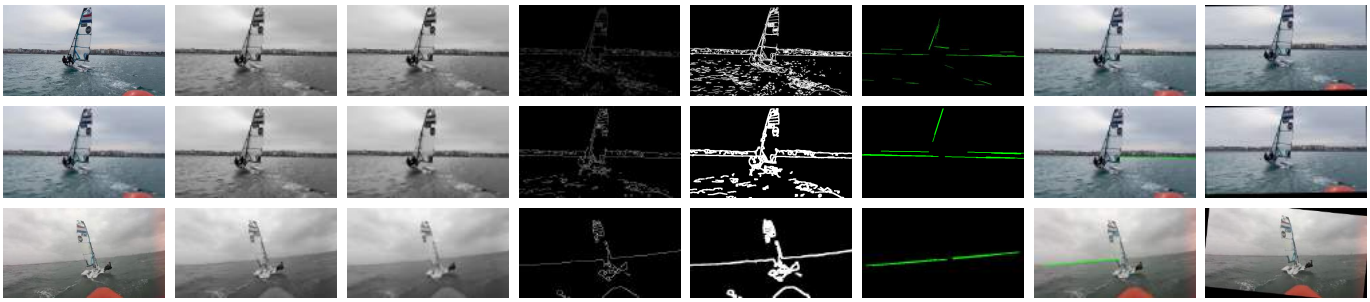


FIGURE A.1: Stabilization pipeline examples with sources videos of 1920x1080 pixels, same video downscaled to 640x430 pixels and another video with a resolution of 432x240 pixels.

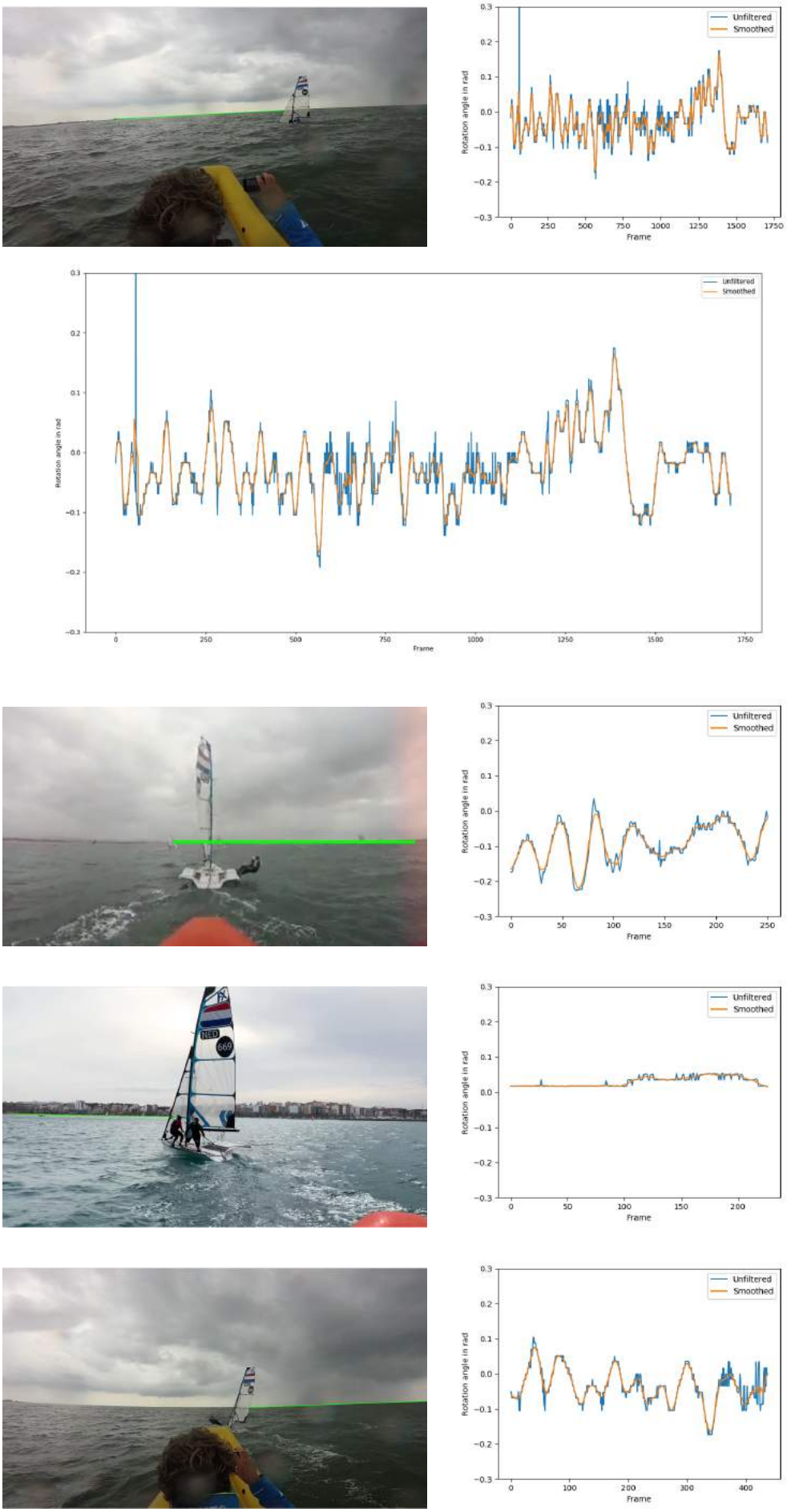
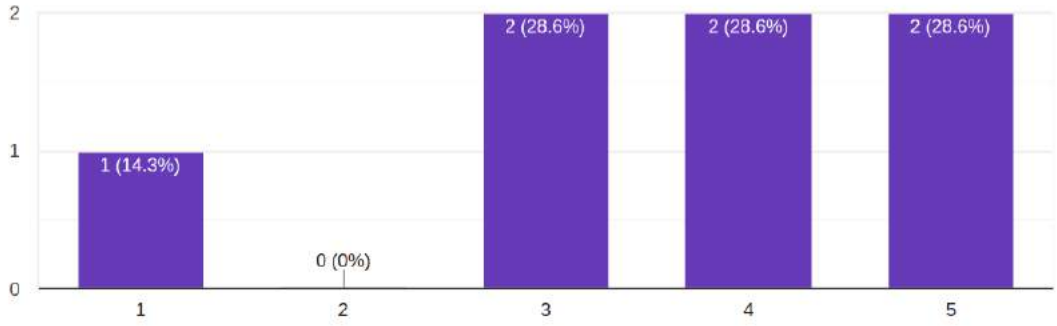


FIGURE A.2: Snapshots of stabilized videos with green line representing detected horizon and accompanying graphs of rotational angle.

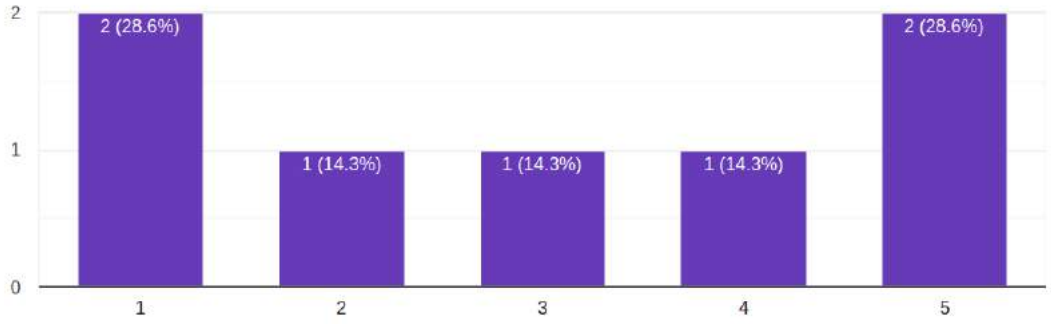
(A) First pair of videos shown, video A being the stabilized version.



(B) Voting results for first pair of videos used in user study.

FIGURE A.3: Video and results of first video pair used in user study.

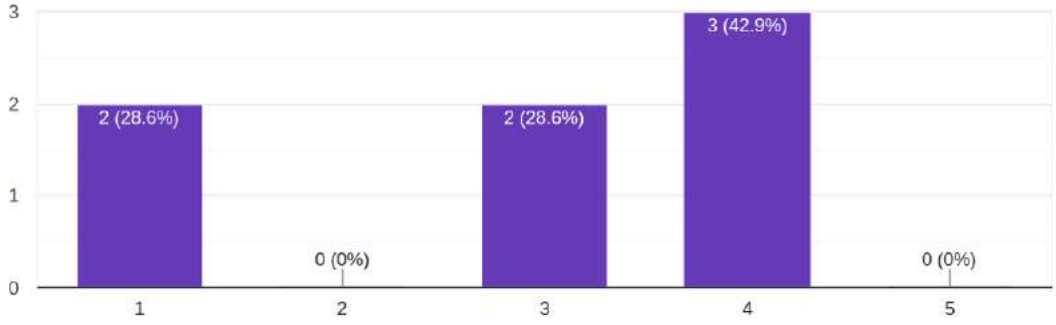
(A) Second pair of videos shown, video 1 being the stabilized version.



(B) Voting results for second pair of videos used in user study.

FIGURE A.4: Video and results of second video pair used in user study.

(A) Third pair of videos shown, video A being the stabilized version.



(B) Voting results for third pair of videos used in user study.

FIGURE A.5: Video and results of third video pair used in user study.

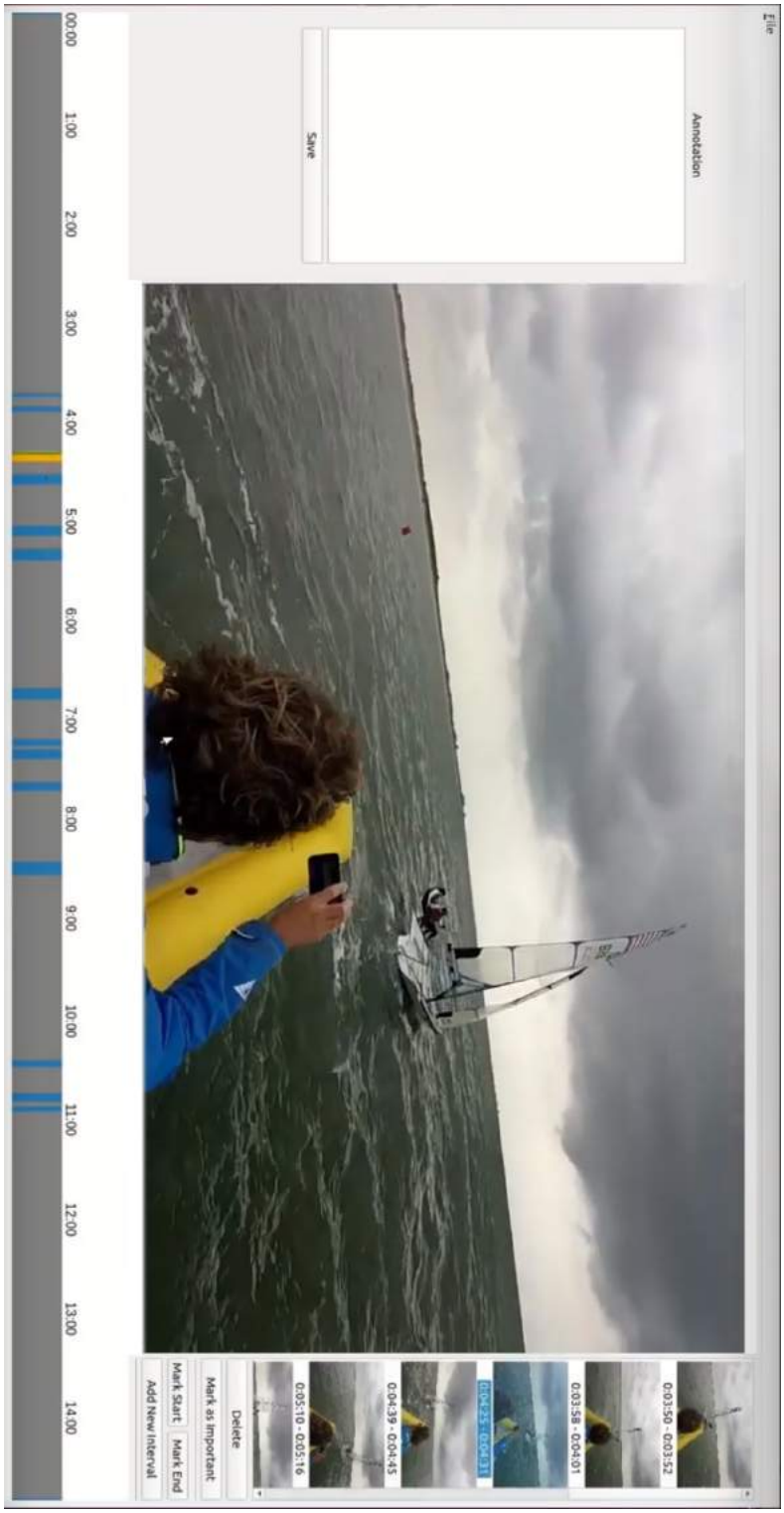


FIGURE A.6: Visual Analysis Framework Interface

Appendix B

User Study Responses

		Which video (1 or A) is easier to analyze? (1) - Strong preference for Video 1 ; (3) - Both videos are equally easy/difficult); (5) - Strong preference for Video A	*Please motivate your choice:*	*Which video (1 or A) is easier to analyze? (1) - Strong preference for Video 1 ; (3) - Both videos are equally easy/difficult); (5) - Strong preference for Video A*	*Which video (1 or A) is easier to analyze? (1) - Strong preference for Video 1 ; (3) - Both videos are equally easy/difficult); (5) - Strong preference for Video A*	*What aspects of this framework, if any, would be useful for coaching?*
2		**4**	Kleine voorkeur voor video A waarbij de horizon recht blijft staan. Maar benieuwd of dit ook prettig kijkt wanneer de omstandigheden erg ruig zijn. Vandaar dat het mij niet mega veel uitmaakt.	**5**	**1**	mark important intervals
3		**3**	Maakt mij weinig uit. Beide filmpjes zijn oké.	**3**	**3**	Geen
4		**4**	Minder beweging in het omliggende beeld; meer stabiliteit maakt het kijken naar details makkelijker	**2**	**3**	notities, thumbnails, mark as important
5		**5**	Video A would be better if surroundings are cropped; really distracting at the moment to have the changing sides; cause those are continuous references for the brain	**1**	**4**	all
6		**3**	Kijkt gelijk	**4**	**4**	alles; met name de tools die het makkelijk maken om makkelijk terug te laten zien op het water op de iPad
7		**5**	Sowieso beter met de vaste horizon. Bewegende horizon wordt veroorzaakt door het bewegen van de rubberboot. Deze bewegingen zijn totaal irrelevant voor de analyze en kunnen zelfs storend zijn.	**1**	**4**	clickable intervals in timeline notes
8		**1**	Stabiel beeld. Kan je beter focussen op de boot	**5**	**1**	All could be useful. How would you store them/archief them? Can you send them to somebody an will they see the notes? Or do you need the software program

FIGURE B.1: Complete responses User study, part 1 of 2.

1	*What features are you missing in this framework?*	*What features are not useful?*	*How likely is it that you would use this in coaching?*	*How useful, in your opinion, is the timeline with marked intervals/manoeuvres in the framework?*
2	geen	?	**2**	**9**
3	Geen details	Te ver weg	**1**	**6**
4	- de actuele tijd: nu alleen tijdstip in de sessie; - een 'add to library' button; zodat we niet alles zomaar opslaan -	-	**4**	**6**
5	don't know yet	none	**4**	**8**
6	nog niks; misschien komen ze naar voren bij gebruik	-	**4**	**7**
7	Quick export (met een druk op de knop) van een bepaald stukje video naar telefoon van coach of sporters. Bluetooth, airdrop, whatsapp oid...	die lijst aan de rechterkant want alles ziet er hetzelfde uit. Je moet het een naam geven wil je het kunnen herkennen.	**3**	**8**
8	Maybe a drawing option? Can you name/label the different clips? Can you zoom? The video is quit far away? If video is high quality maybe you can add a zoom function that would follow the boat?	All ok I suppose	**4**	**8**

FIGURE B.2: Complete responses User study, part 2 of 2.

Bibliography

[1] S. Barris and C. Button, "A review of vision-based motion analysis in sport", *Sports Medicine*, vol. 38, no. 12, pp. 1025–1043, 2008, ISSN: 1179-2035. DOI: 10.2165/00007256-200838120-00006. [Online]. Available: https://doi.org/10.2165/00007256-200838120-00006.

[2] L. J. Nelson, P. Potrac, and R. Groom, "Receiving video-based feedback in elite ice-hockey: A player's perspective", *Sport, Education and Society*, vol. 19, no. 1, pp. 19–40, 2014. DOI: 10.1080/13573322.2011.613925. eprint: https://doi.org/10.1080/13573322.2011.613925. [Online]. Available: https://doi.org/10.1080/13573322.2011.613925.

[3] D. G. Liebermann, L. Katz, M. D. Hughes, R. M. Bartlett, J. McClements, and I. M. Franks, "Advances in the application of information technology to sport performance", *Journal of Sports Sciences*, vol. 20, no. 10, pp. 755–769, 2002, PMID: 12363293. DOI: 10.1080/026404102320675611. eprint: https://doi.org/10.1080/026404102320675611. [Online]. Available: https://doi.org/10.1080/026404102320675611.

[4] R. Bartlett, "Performance analysis: Can bringing together biomechanics and notational analysis benefit coaches?", *International Journal of Performance Analysis in Sport*, vol. 1, no. 1, pp. 122–126, 2001. DOI: 10.1080/24748668.2001.11868254. eprint: https://doi.org/10.1080/24748668.2001.11868254. [Online]. Available: https://doi.org/10.1080/24748668.2001.11868254.

[5] J. C. Ives, W. F. Straub, and G. A. Shelley, "Enhancing athletic performance using digital video in consulting", *Journal of Applied Sport Psychology*, vol. 14, no. 3, pp. 237–245, 2002. DOI: 10.1080/10413200290103527. eprint: https://doi.org/10.1080/10413200290103527. [Online]. Available: https://doi.org/10.1080/10413200290103527.

[6] P. O'Donoghue, "The use of feedback videos in sport", *International Journal of Performance Analysis in Sport*, vol. 6, pp. 1–14, Nov. 2006. DOI: 10.1080/24748668.2006.11868368.

[7] P. Cunningham and T. Hale, "Physiological responses of elite laser sailors to 30 minutes of simulated upwind sailing", *Journal of Sports Sciences*, vol. 25, no. 10, pp. 1109–1116, 2007, PMID: 17613734. DOI: 10.1080/02640410601165668. eprint: https://doi.org/10.1080/02640410601165668. [Online]. Available: https://doi.org/10.1080/02640410601165668.

[8] M. Blackburn, "Physiological responses to 90 min of simulated dinghy sailing", *Journal of Sports Sciences*, vol. 12, no. 4, pp. 383–390, 1994, PMID: 7932949. DOI: 10.1080/02640419408732185. eprint: https://doi.org/10.1080/02640419408732185. [Online]. Available: https://doi.org/10.1080/02640419408732185.

[9] T. Gale and J. Walls, "Development of a sailing dinghy simulator", *SIMULATION*, vol. 74, no. 3, pp. 167–179, 2000. DOI: 10.1177/003754970007400304. eprint: https://doi.org/10.1177/003754970007400304. [Online]. Available: https://doi.org/10.1177/003754970007400304.

[10] J. P. Pluijms, R. Cañal-Bruland, M. J. Hoozemans, and G. J. Savelsbergh, "Visual search, movement behaviour and boat control during the windward mark rounding in sailing", *Journal of sports sciences*, vol. 33, no. 4, pp. 398–410, 2015.

[11] S Legg, H Mackie, and P Smith, "Temporal patterns of physical activity in olympic dinghy racing", *Journal of sports medicine and physical fitness*, vol. 39, no. 4, p. 315, 1999.

[12] H Mackie, "Useful biomechanics for sailing: Development of technique analysis protocol for europe and laser sailors", in *Human performance in sailing conference proceedings: incorporating the 4th European Conference on Sailing Sports Science and Sports Medicine and the 3rd Australian Sailing Science Conference. Palmerston North, New Zealand: Massey University*, 2003, pp. 71–75.

[13] C. Boehm, R. Brehm, J. Meyer, L. Duggen, and K. Graf, "A measurement system for performance monitoring on small sailing dinghies", 2013.

[14] S. Honey and K. Milnes, "The annotated america's cup", *Spectrum, IEEE*, vol. 50, pp. 36–42, Sep. 2013. DOI: 10.1109/MSPEC.2013.6587187.

[15] H. Zhang, P. Yin, X. Zhang, and X. Shen, "A robust adaptive horizon recognizing algorithm based on projection", *Transactions of the Institute of Measurement and Control*, vol. 33, no. 6, pp. 734–751, 2011. DOI: 10.1177/0142331209342201. eprint: https://doi.org/10.1177/0142331209342201. [Online]. Available: https://doi.org/10.1177/0142331209342201.

[16] Y.-S. Yao, P. Burlina, R. Chellappa, and T.-H. Wu, "Electronic image stabilization using multiple visual cues", in *Proceedings., International Conference on Image Processing*, IEEE, vol. 1, 1995, pp. 191–194.

[17] M. Schwendeman and J. Thomson, "A horizon-tracking method for shipboard video stabilization and rectification", *Journal of Atmospheric and Oceanic Technology*, vol. 32, no. 1, pp. 164–176, 2015.

[18] H. Cao and J. Zhang, "Video stabilizing and tracking by horizontal line for maritime cruise ship", in *2007 IEEE International Conference on Control and Automation*, IEEE, 2007, pp. 1202–1206.

[19] X. Yantai and W. Tao, "A new horizon electronic image stabilization algorithm based on svm", in *2010 International Conference on Optoelectronics and Image Processing*, IEEE, vol. 1, 2010, pp. 59–63.

[20] L. Steccanella, D. Bloisi, J. Blum, and A. Farinelli, "Deep learning waterline detection for low-cost autonomous boats", in *International Conference on Intelligent Autonomous Systems*, Springer, 2018, pp. 613–625.

[21] C. Morimoto and R. Chellappa, "Fast electronic digital image stabilization", in *Proceedings of 13th International Conference on Pattern Recognition*, IEEE, vol. 3, 1996, pp. 284–288.

[22] L. Wen, Z. Yingjun, and Y. Xuefeng, "A feature-based method for shipboard video stabilization", in *2019 IEEE 2nd International Conference on Electronic Information and Communication Technology (ICEICT)*, IEEE, 2019, pp. 315–322.

[23] D. D. Morris, B. R. Colonna, and F. D. Snyder, "Image-based motion stabilization for maritime surveillance", in *Image Processing: Algorithms and Systems V*, International Society for Optics and Photonics, vol. 6497, 2007, 64970F.

[24] X. Peng, R. Liu, Y. L. Murphey, S. Stent, and Y. Li, "Driving maneuver detection via sequence learning from vehicle signals and video images", in *2018 24th International Conference on Pattern Recognition (ICPR)*, IEEE, 2018, pp. 1265–1270.

[25] H. Li, J. Tang, S. Wu, Y. Zhang, and S. Lin, "Automatic detection and analysis of player action in moving background sports video sequences", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 3, pp. 351–364, 2010, cited By 47. DOI: 10.1109/TCSVT.2009.2035833. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-77749310224&doi=10.1109%2fTCSVT.2009.2035833&partnerID=40&md5=35b8f950e8fb22d658436a2523f90774.

[26] S. Barris and C. Button, "A review of vision-based motion analysis in sport", *Sports Medicine*, vol. 38, no. 12, pp. 1025–1043, 2008, cited By 134. DOI: 10.2165/00007256-200838120-00006. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-56749103680&doi=10.2165%2f00007256-200838120-00006&partnerID=40&md5=c59a0e47e9847f91557c45a5c4338a6b.

[27] G. Zhu, C. Xu, Q. Huang, Y. Rui, S. Jiang, W. Gao, and H. Yao, "Event tactic analysis based on broadcast sports video", *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 49–67, 2009, cited By 81. DOI: 10.1109/TMM.2008.2008918.

[28] C. B. Santiago, A. Sousa, M. L. Estriga, L. P. Reis, and M. Lames, "Survey on team tracking techniques applied to sports", in *2010 International Conference on Autonomous and Intelligent Systems, AIS 2010*, 2010, pp. 1–6. DOI: 10.1109/AIS.2010.5547021.

[29] T. D'Orazio and M. Leo, "A review of vision-based systems for soccer video analysis", *Pattern Recognition*, vol. 43, no. 8, pp. 2911 – 2926, 2010, ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2010.03.009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320310001299.

[30] J. J. Athanesious and P Suresh, "Systematic survey on object tracking methods in video", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 1, no. 8, pp. 242–247, 2012.

[31] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey", *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[32] H. S. Parekh, D. G. Thakore, and U. K. Jaliya, "A survey on object detection and tracking methods", *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 2, pp. 2970–2979, 2014.

[33] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey", *arXiv preprint arXiv:1809.02165*, 2018.

[34] M. Yazdi and T. Bouwmans, "New trends on moving object detection in video images captured by a moving camera: A survey", *Computer Science Review*, vol. 28, pp. 157–177, 2018.

[35] C. Novak and S. Shafer, "Color edge detection", in *Image Understanding Workshop, vol. I*, 1987.

[36] ITU, "Studio encoding parameters of digital television for standard 4: 3 and wide-screen 16: 9 aspect ratios", 2011. [Online]. Available: `https://www.itu.int/rec/R-REC-BT.601-7-201103-I/en`.

[37] B. Weiss, "Fast median and bilateral filtering", *Acm Transactions on Graphics (TOG)*, vol. 25, no. 3, pp. 519–526, 2006.

[38] J. Canny, "A computational approach to edge detection", *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.

[39] N. Jawas and N. Suciati, "Image inpainting using erosion and dilation operation", *International Journal of Advanced Science and Technology*, vol. 51, pp. 127–134, 2013.

[40] J. Matas, C. Galambos, and J. Kittler, "Robust detection of lines using the progressive probabilistic hough transform", *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 119 –137, 2000, ISSN: 1077-3142. DOI: `https://doi.org/10.1006/cviu.1999.0831`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S1077314299908317`.

[41] M. A. Qureshi, M. Deriche, A. Beghdadi, and A. Amin, "A critical survey of state-of-the-art image inpainting quality assessment metrics", *Journal of Visual Communication and Image Representation*, vol. 49, pp. 177 –191, 2017, ISSN: 1047-3203. DOI: `https://doi.org/10.1016/j.jvcir.2017.09.006`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S1047320317301803`.

[42] C. Hughes, M. Glavin, E. Jones, and P. Denny, "Review of geometric distortion compensation in fish-eye cameras", Jul. 2008, pp. 162 –167. DOI: `10.1049/cp:20080656`.

[43] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review", *IEEE transactions on neural networks and learning systems*, 2019.

[44] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications", *arXiv preprint arXiv:1704.04861*, 2017.

[45] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7310–7311.

[46] X. Wu, D. Sahoo, and S. C. H. Hoi, *Recent advances in deep learning for object detection*, 2019. arXiv: 1908.03673 [cs.CV].

[47] 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 *European conference on computer vision*, Springer, 2014, pp. 740–755.

[48] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of exemplar-svms for object detection and beyond", in *ICCV*, 2011.

[49] R. E. Kalman, "A new approach to linear filtering and prediction problems", *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[50] A. Lukezic, T. Vojir, L. Cehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability", in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[51] F. Bergholm, "Edge focusing", *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 726–741, 1987.

[52] A. P. Witkin, "Scale-space filtering", in *Readings in Computer Vision*, Elsevier, 1987, pp. 329–332.

[53] B. M. H. Romeny, *Front-end vision and multi-scale image analysis: multi-scale computer vision theory and applications, written in mathematica*. Springer Science & Business Media, 2008, vol. 27.

[54] M. H. Kutner, C. J. Nachtsheim, J. Neter, W. Li, *et al.*, *Applied linear statistical models*. McGraw-Hill Irwin New York, 2005, vol. 5.

[55] W. O. Galitz, *The essential guide to user interface design: an introduction to GUI design principles and techniques*. John Wiley & Sons, 2007.

[56] J. Korhonen and J. You, "Peak signal-to-noise ratio revisited: Is simple beautiful?", in *2012 Fourth International Workshop on Quality of Multimedia Experience*, 2012, pp. 37–38. DOI: 10.1109/QoMEX.2012.6263880.

[57] R. Likert, "A technique for the measurement of attitudes.", *Archives of psychology*, 1932.