Università degli Studi di Padova

Universitat Politècnica de Catalunya

ETSEIB

UPC

THESIS DEFENSE

Real-Time Gaze Depth Estimation Using a Wearable Eye Tracking Device

Presented by Luca Secchieri

Supervisor: Manel Frigola Bourlon

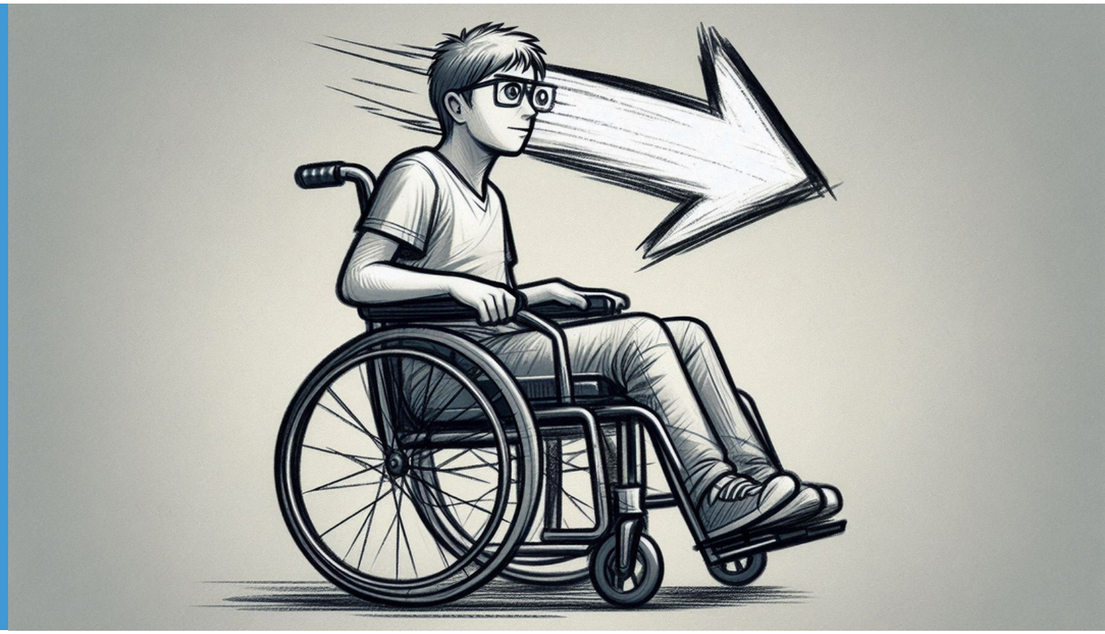
A.Y. 2023/24

OVERVIEW

INTRODUCTION

This project is part of a larger initiative focused on **developing a robotized wheelchair for individuals with disabilities.**

The **primary objective** of this thesis is to develop a method for **estimating gaze depth** using images captured by Tobii Pro Glasses 3.



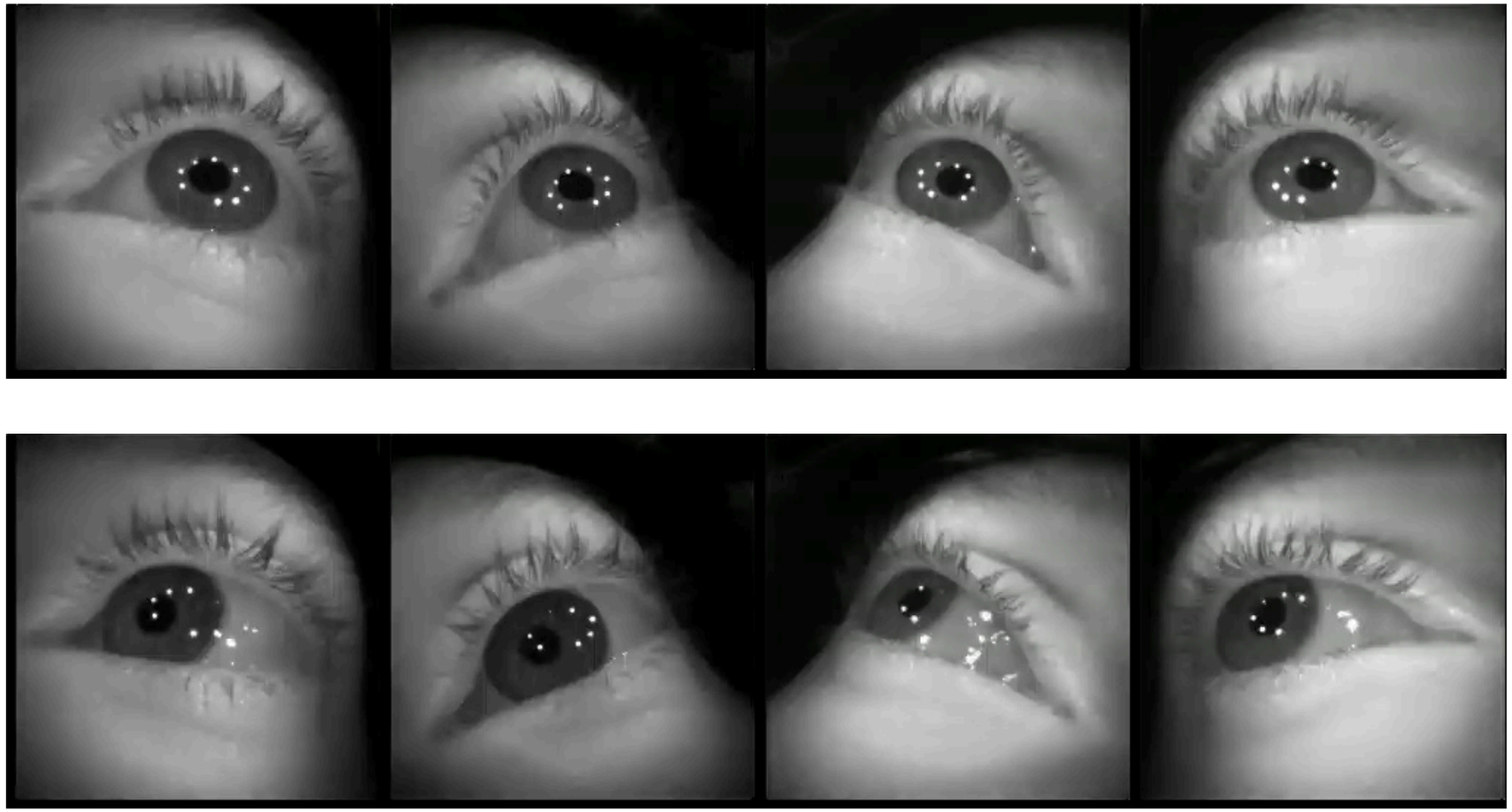
EQUIPMENT

Tobii Pro Glasses 3

A state-of-the-art wearable eye tracker, the Tobii Pro Glasses 3, was utilized to collect real-world data. Its sophisticated technology facilitates comprehensive data collection, which is critical for our research and analysis.

KEY FEATURES

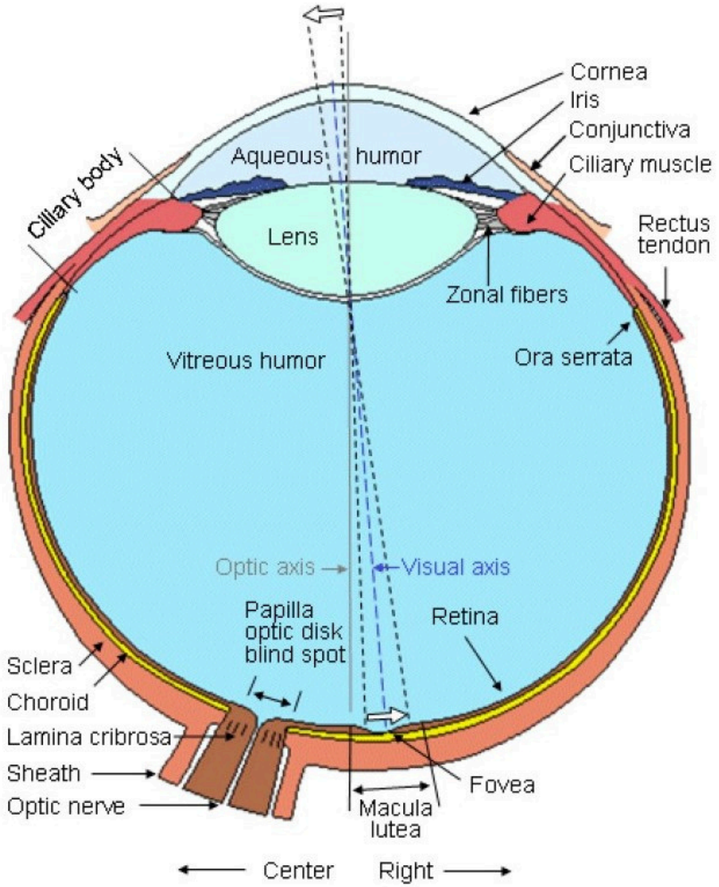
SOME EXAMPLES



THE HUMAN EYE

3D Vision

- Both monocular and binocular cues.
- Depth perception is based on:
  - Accomodation
  - Miosis
  - Vergence



EYE VERGENCE

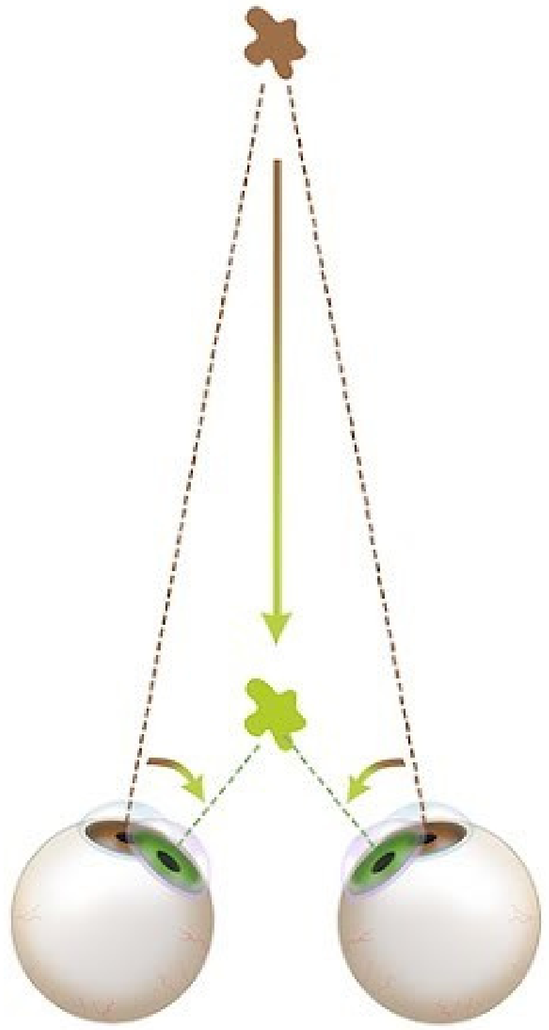
Vergence involves the simultaneous movement of both eyes in opposite directions to maintain single binocular vision.

Convergence

To focus on nearby objects, our eyes turn inward in a coordinated movement called convergence.

Divergence

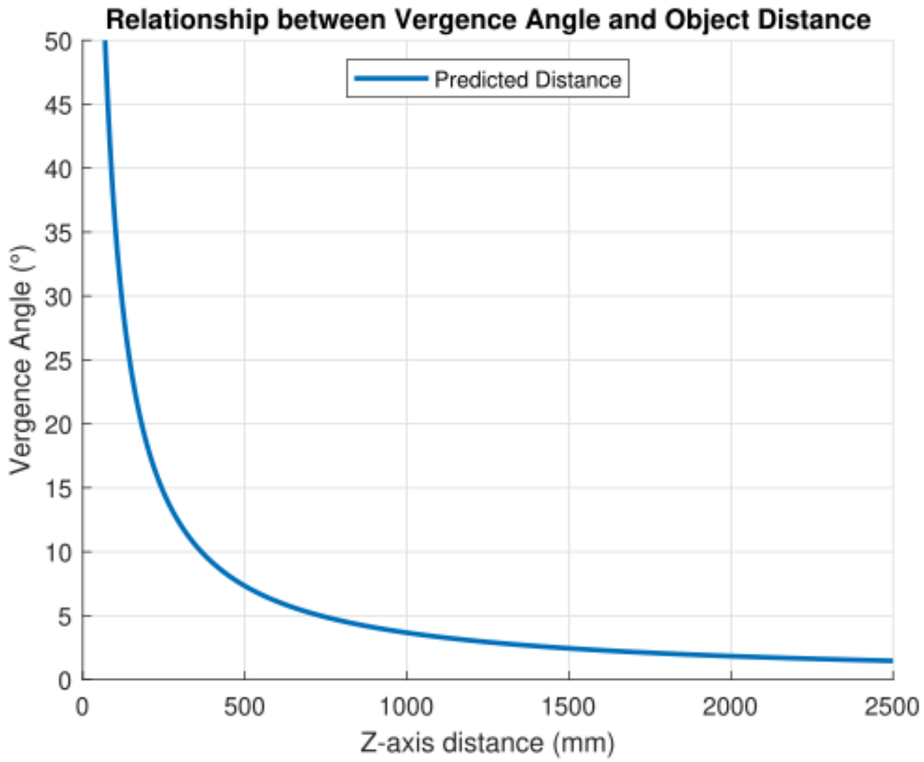
When shifting focus to distant objects, the eyes rotate outward in a process known as divergence.



EYE VERGENCE ANGLE - DISTANCE

Theoretical vergence angle as a function of distance

$$\theta = 2arctan\left(\frac{d}{2z}\right)$$



PROJECT EXECUTION STEPS

Phase 1

Creation of a robust pupil detection procedure to extract their coordinates.

Phase 2

Development of a regression model capable of mapping pupil coordinates into a gaze depth estimation measure.

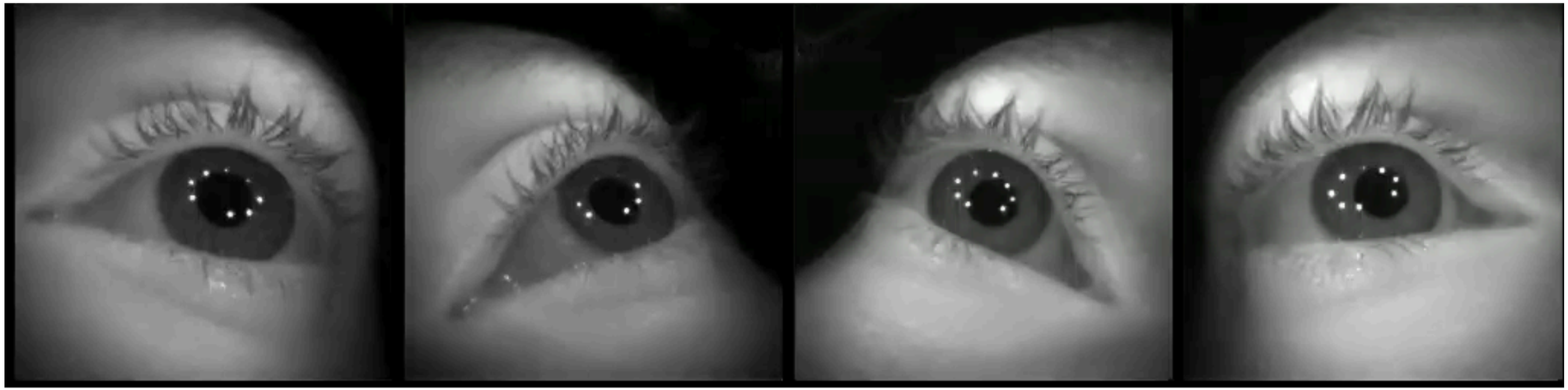
Phase 3

Real-time video decoding to obtain a continuos estimation of the gaze depth.

PUPIL DETECTION

Objective

Extracting the (X,Y) coordinates of the pupils' center from the frames obtained through Tobii Pro Glasses 3.



1024x256

PUPIL DETECTION PROCEDURES

Binarization

The first method requires the binarization of the image to separate the pupil from the rest of the eye.
Implemented in Python using OpenCV.

Hough

The second approach leverages the Hough transform to detect circles, corresponding to the pupils, in the image.
Implemented in Python using OpenCV.

Matlab

The last approach was implemented in Matlab.
It revolves around the *imfindcircles* function, which uses a Circular Hough Transform (CHT) based algorithm for finding circles in images.

BINARIZATION METHOD

Pre-Processing

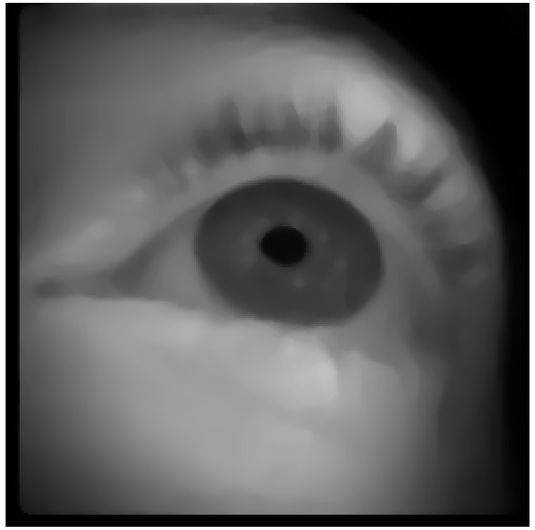
Median blur and Erosion.
Remove infrared reflections.

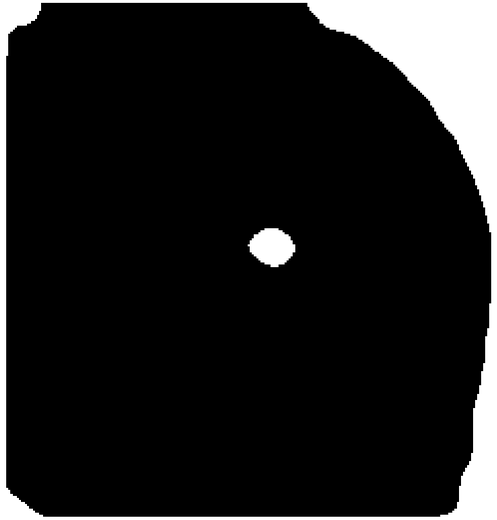
Binarization

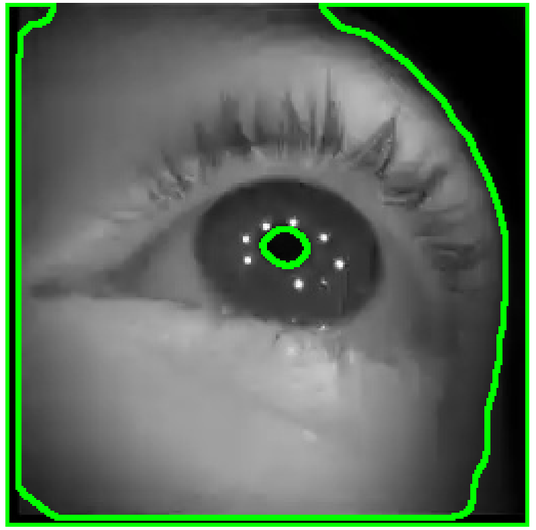
Threshold set at TH.
If i < TH, pixel is black.
If i >= TH, pixel is white.

Contour extraction

Using cv2.findContours(),
Then to detect the pupil
cv2.minEnclosingCircle().



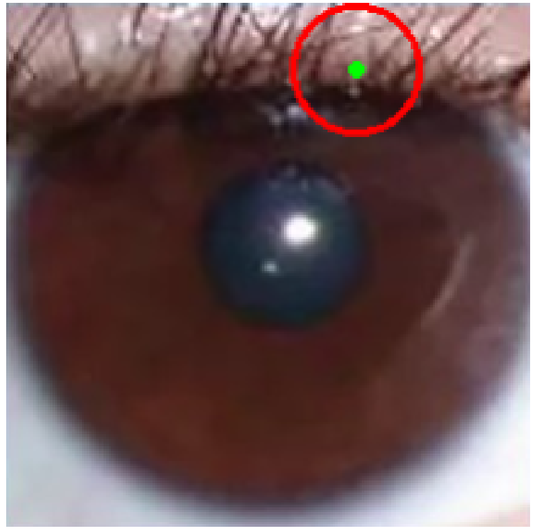




ADAPTIVE THRESHOLDING

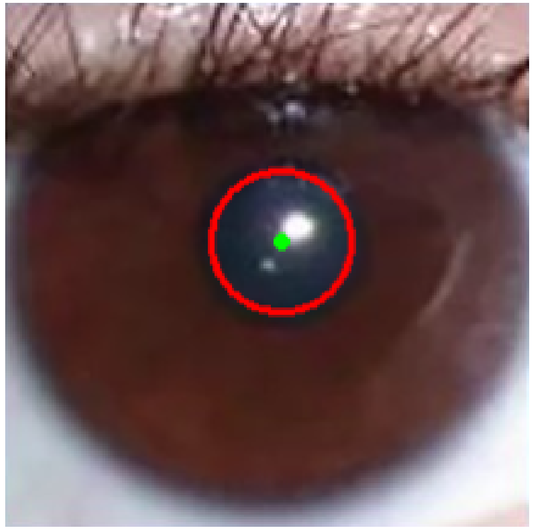
Fixed Threshold

Fails when there is low contrast between the iris and the pupil (dark eyes).



Adaptive Threshold

Dynamically adjusts the threshold value for different regions of the image.



HOUGH TRANSFORM METHOD

Pre-Processing

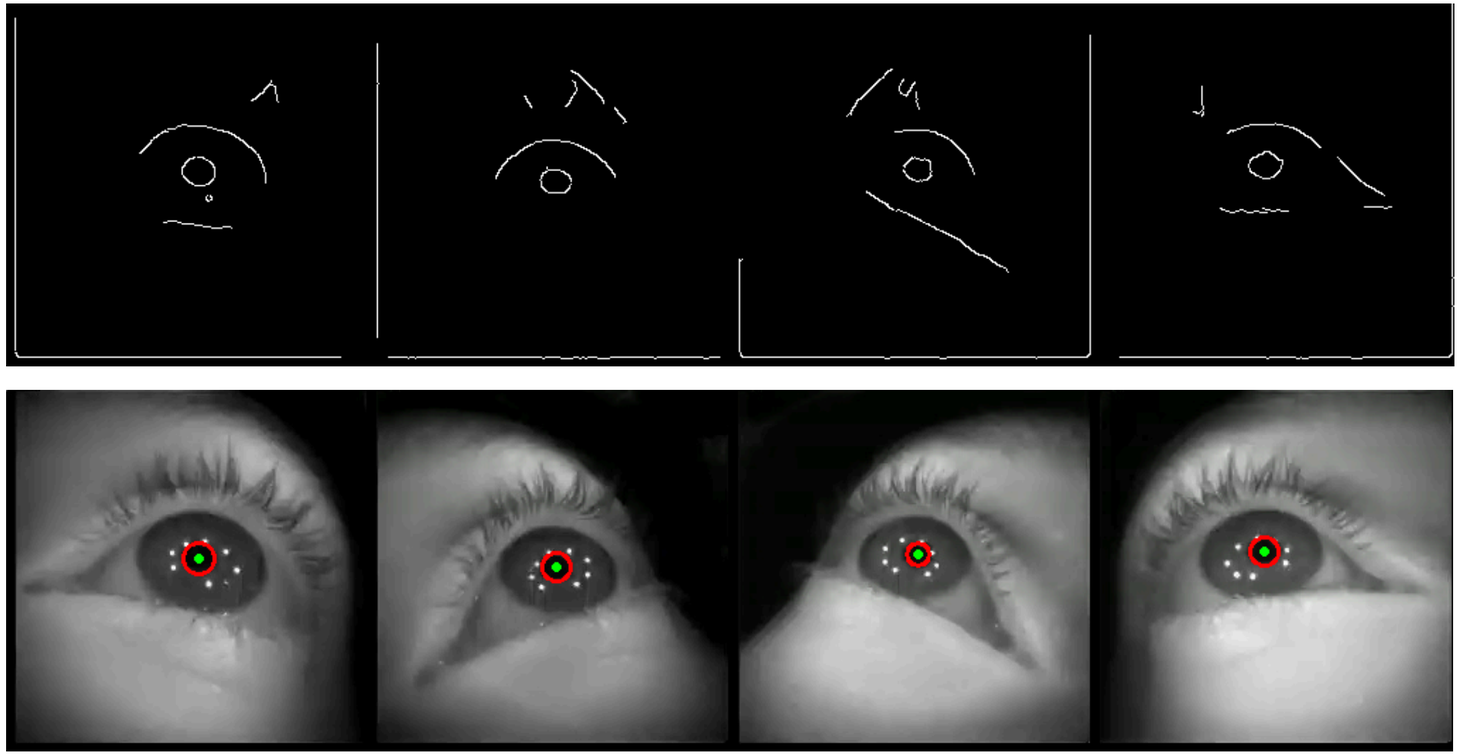
Again a median blur filter, to remove the infrared reflections.

Canny Edge Extraction

Detects edges by looking for areas of high gradient magnitude.

Hough Circle Transform

HCT algorithm is used to detect circular shapes in the edge-detected image.



DETECTION USING MATLAB PRIMITIVES

Pre-Processing

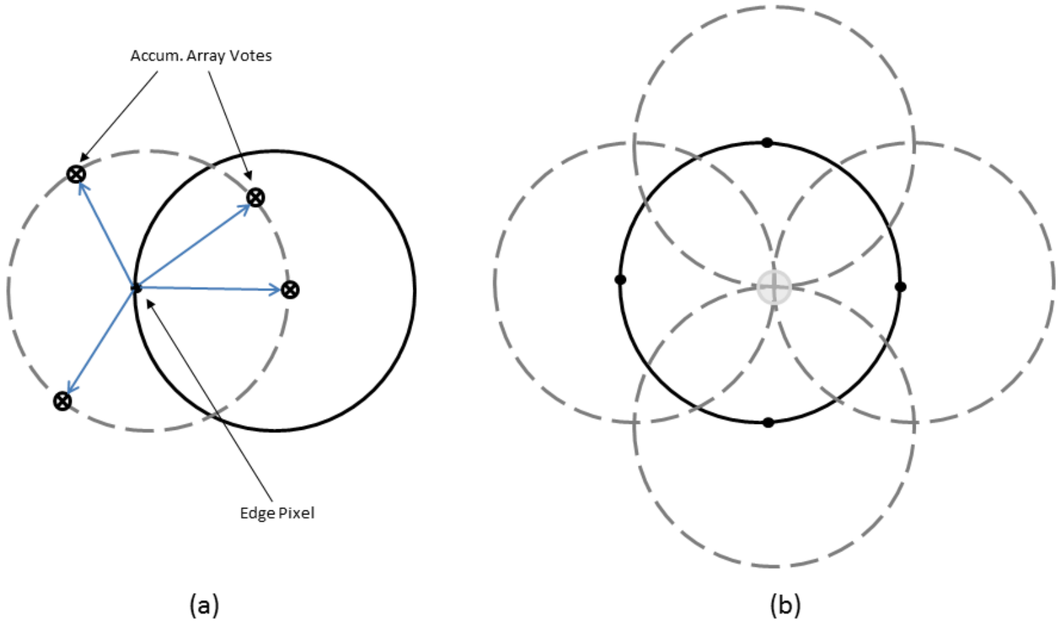
Gaussian blur filtering. Morphological opening to remove infrared reflections.

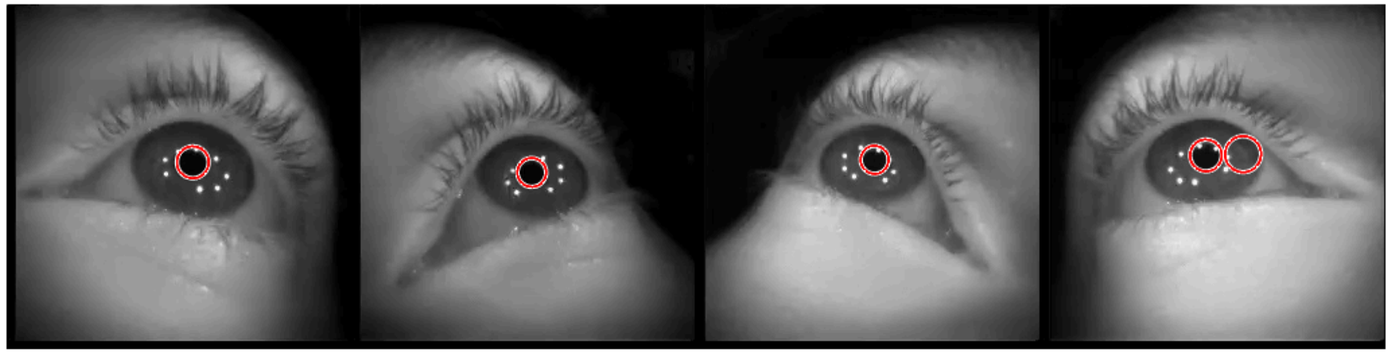
Circle Detection

imfindcircles() primitive that utilizes the Circular Hough Transform.

Handling Detections

Only the circle detected with the highest value in the accumulator is kept.

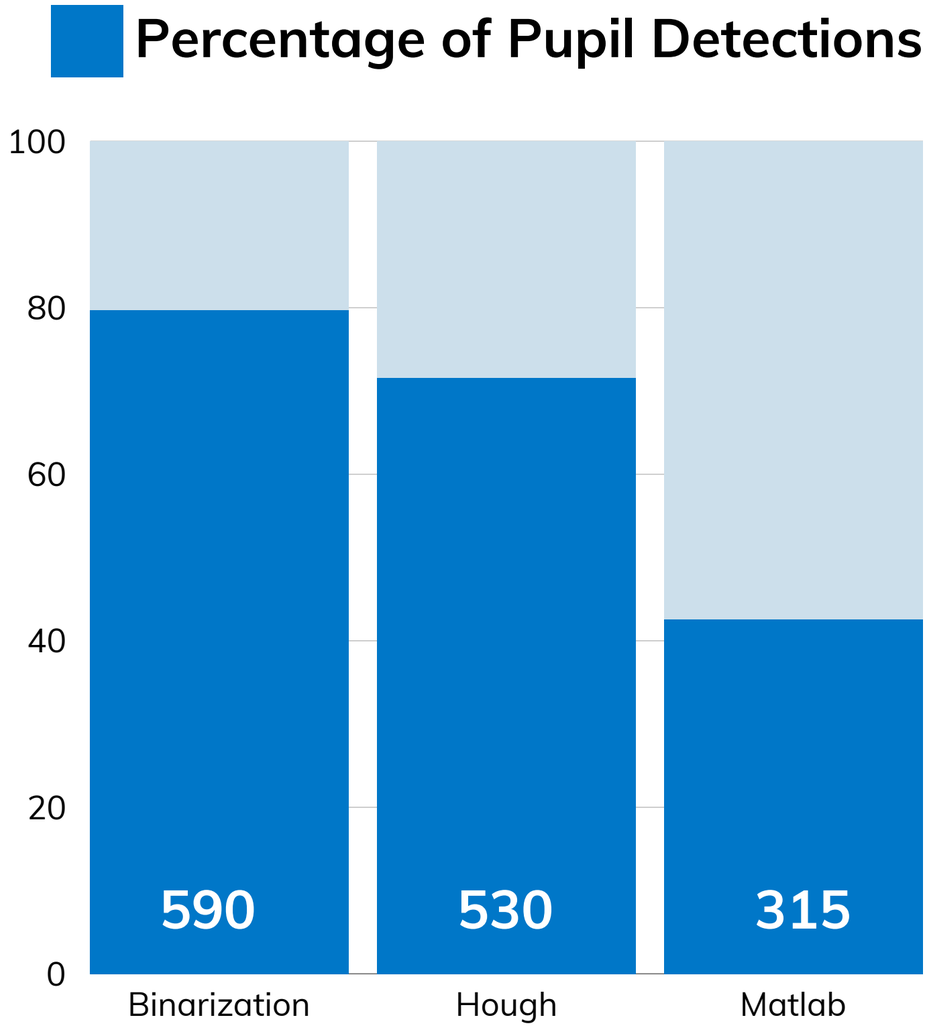




FIRST EXPERIMENTAL SETUP

- From 50cm to 350cm.
- Intervals of 25cm between each target.
- All targets positioned frontally aligned with the observers' eyes.
- Head position was fixed during the collection of the samples.
- 13 videos were captured in total.

PUPIL DETECTION RESULTS

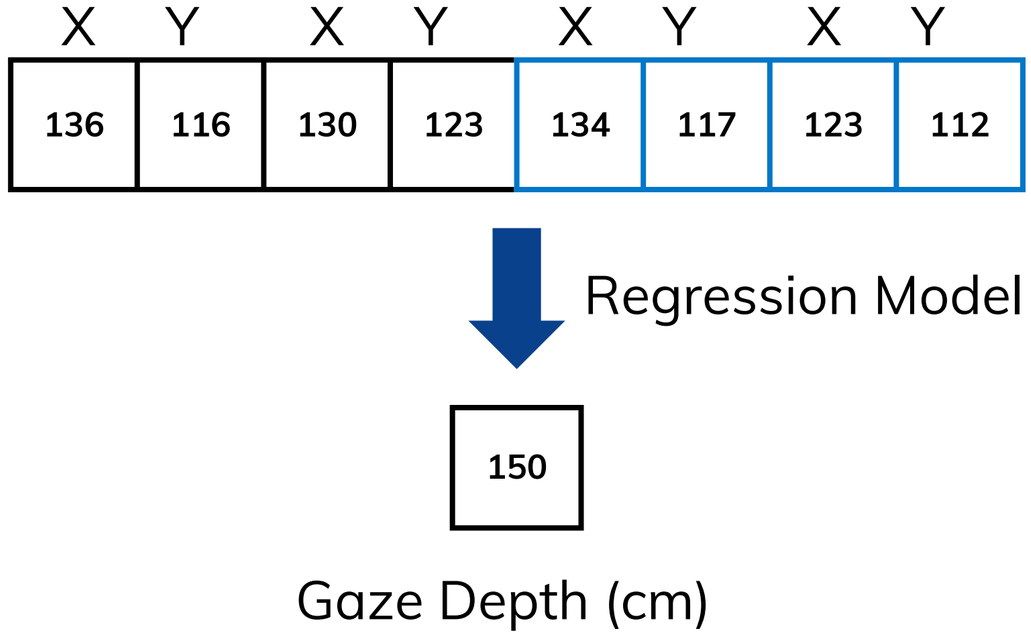


- Initial number of frames is 740
- Binarization has the best overall performance.
- Hough method is effective at detecting pupils but it’s inconsistent in the position of the pupil between consecutive frames.
- Matlab method has the highest rate of missed detections.

REGRESSION MODEL

The objective of the following part is to develop a regression model capable of estimating gaze depth using pupil position data.

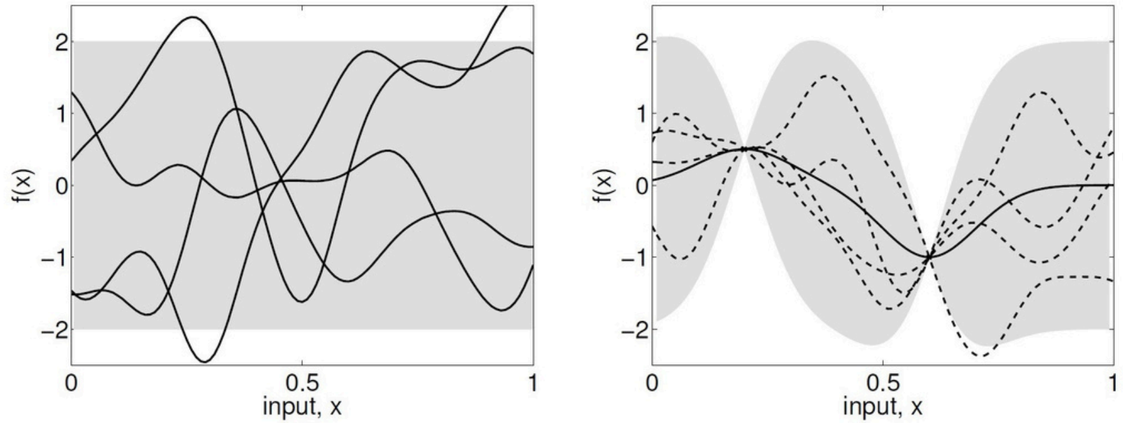
MATLAB's regression learner app was employed to create, train, and evaluate various models.



GAUSSIAN REGRESSION MODELS

Gaussian Process Regression

- Specific application of a GP used for regression tasks.
- Specified by mean m(x) and covariance k(x,x’).
- Assumes the data points follow a Gaussian distribution.



Key Features

Prior

Kernel

Posterior

Prediction

MODEL RESULTS

1. Binarization Dataset
2. Matlab Dataset
3. Second Experiment

Linearizing the Dataset

Given that the relationship between eye vergence and object distance is exponential, and MSE penalizes large errors, we can linearize the input data:

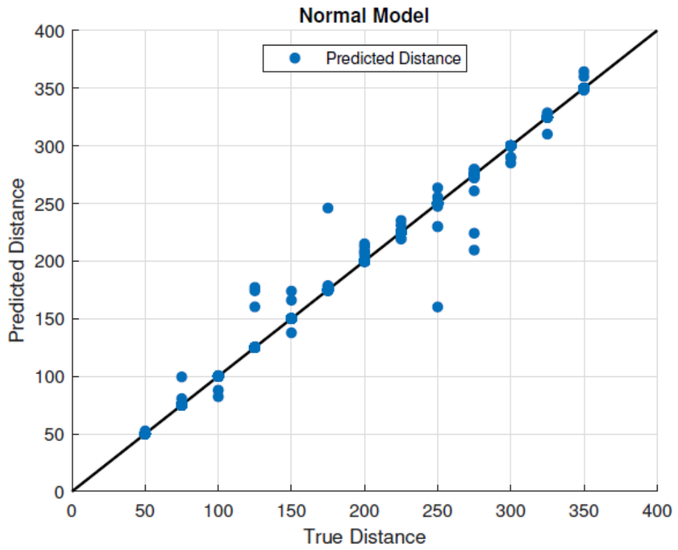
$$d' = \log(d)$$

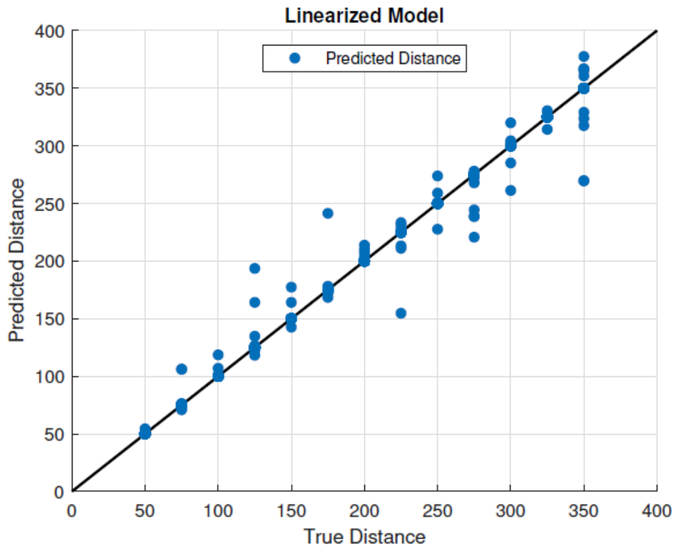
BINARIZATION DATASET

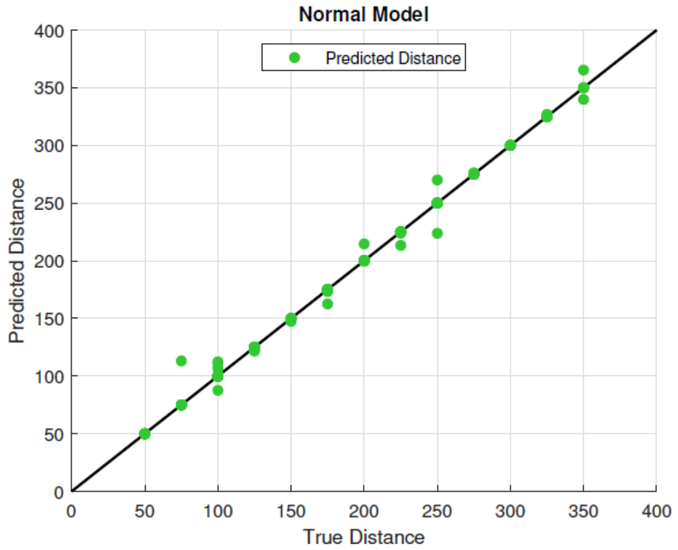
- Trained with 590 samples.
- Best Results obtained with:
  - Rational Quadratic Kernel
  - Exponential Kernel
- Both standard and linearized datasets were tested.

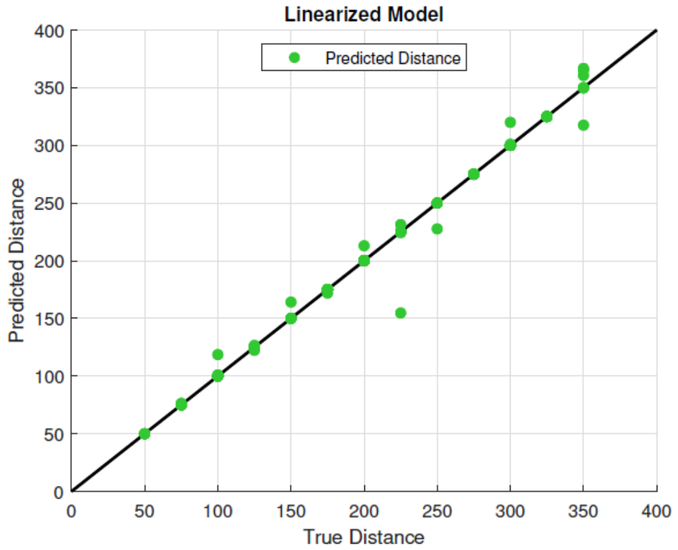
Regression Model	RMSE (k = 5)		RMSE (k = 10)	
	Validation	Test	Validation	Test
Rational quadratic GPR	22.84	17.30	20.81	24.01
Exponential GPR	23.36	19.03	21.53	23.94
Lin. Rational quadratic GPR	27.84	25.46	25.43	28.89
Lin. Exponential GPR	27.88	28.41	26.07	29.45

BINARIZATION DATASET RESULTS







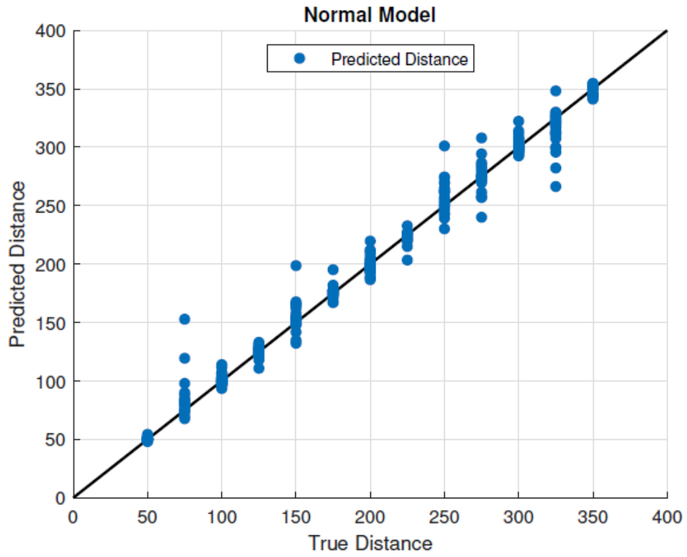


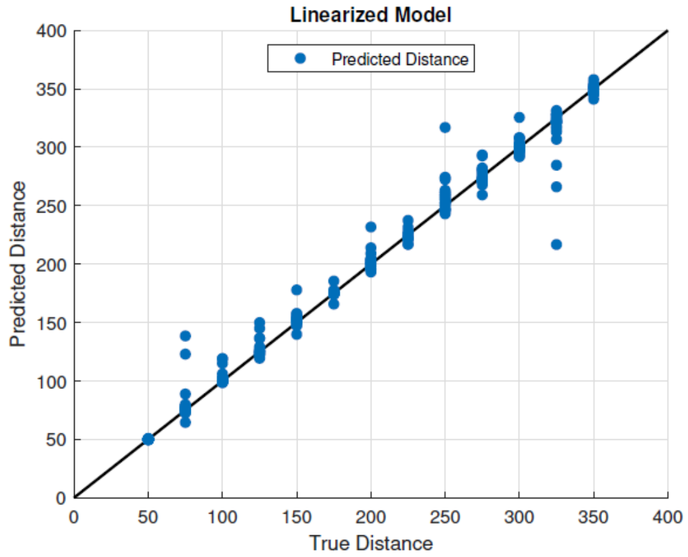
MATLAB DATASET

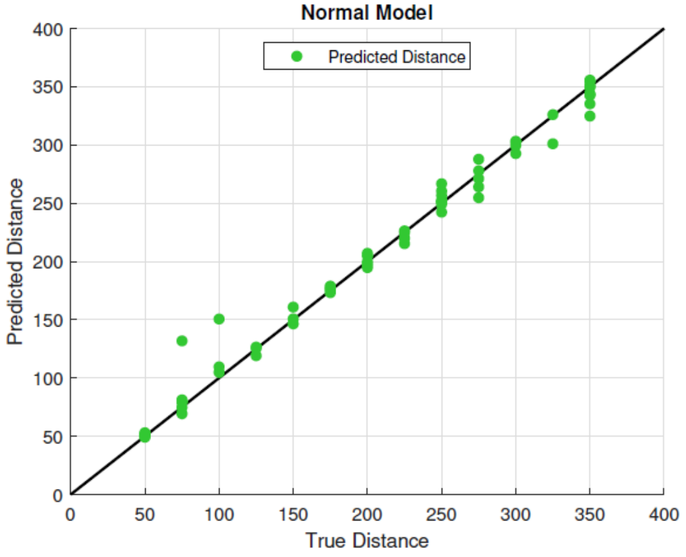
- Trained with 315 samples.
- Best Results obtained with:
  - Rational Quadratic Kernel
  - Matern 5/2 Kernel
- Both standard and linearized datasets were tested.

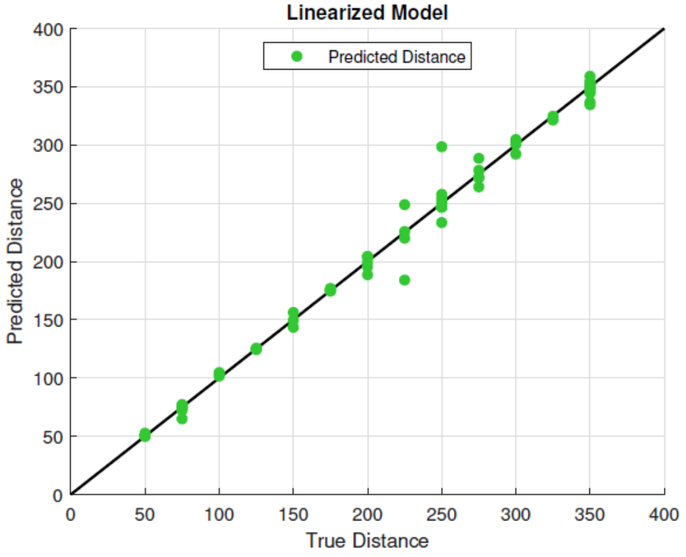
Regression Model	RMSE ($k = 5$)		RMSE (k = 10)	
	Validation	Test	Validation	Test
Rational quadratic GPR	27.01	21.01	27.58	26.79
Matern 5/2 GPR	26.91	21.26	27.74	23.86
Lin. Rational quadratic GPR	31.74	26.26	26.53	23.48
Lin. Matern 5/2 GPR	30.8	26.83	26.04	28.77

MATLAB DATASET RESULTS









SECOND EXPERIMENTAL SETUP

- Targets at 100, 200, 400 cm.
- All targets were still positioned frontally aligned with the observers' eyes.
- The observer was rotated left and right (around 25°) on the chair while keeping the gaze fixed on the target, thus exploiting the vestibulo-ocular reflex.
- 9 videos were captured in total.

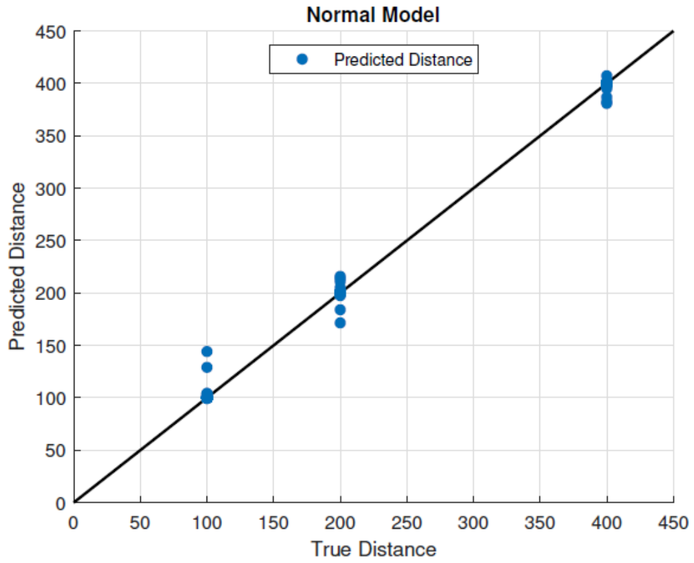
SECOND DATASET

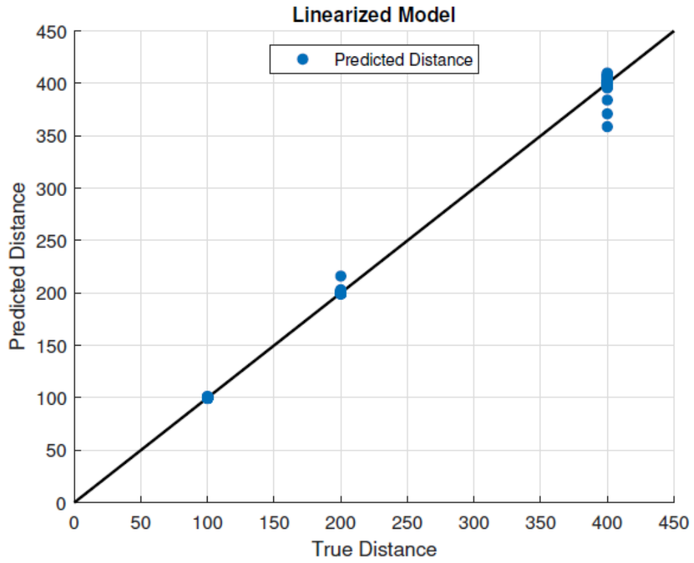
- Trained with 376 samples.
- Best performance: Rational Quadratic Kernel

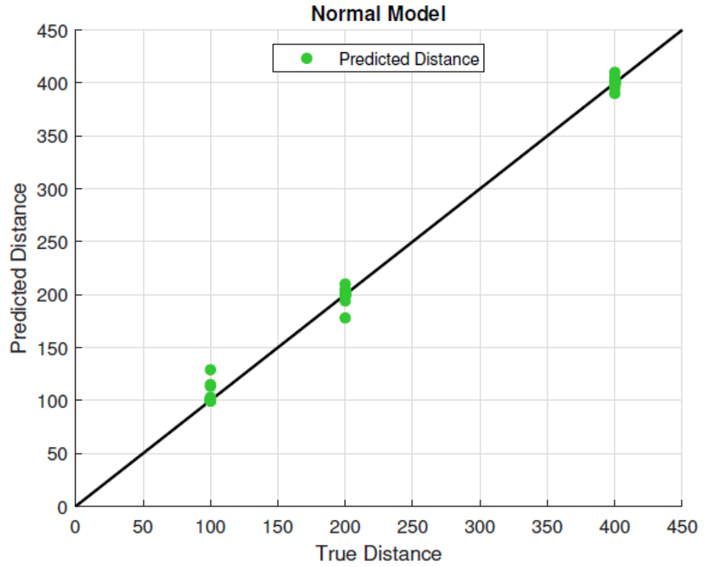
RMSE	Validation	Test
Normal	12.62 cm	8.43 cm
Linearized	14.67 cm	9.13 cm

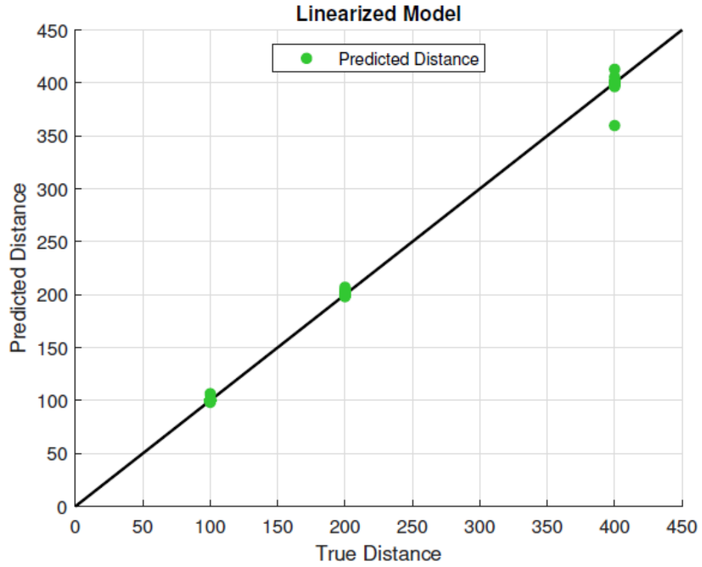
Target Distance	Target Relative Position		
	Left	Frontal	Right
1 meter	33	45	37
2 meter	40	34	36
4 meter	44	50	57

SECOND DATASET RESULTS









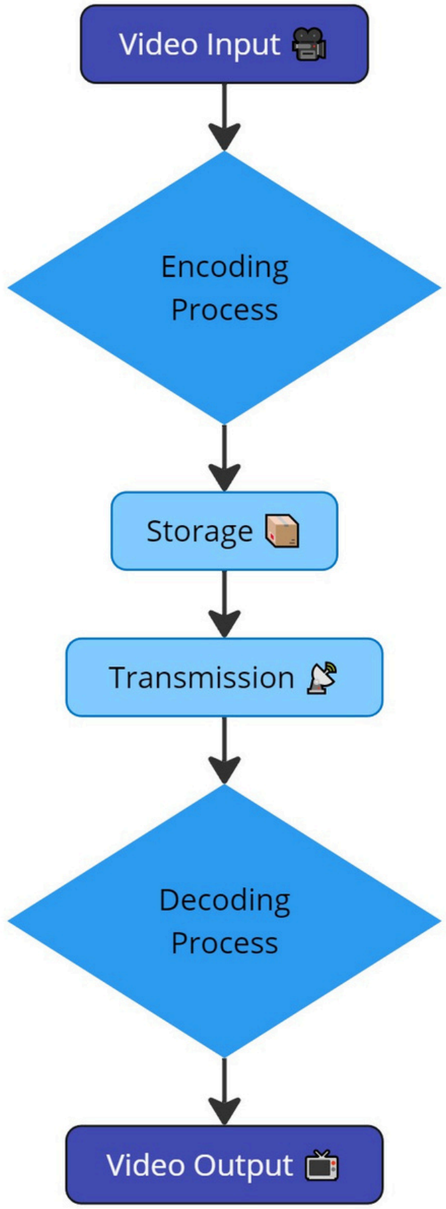
REAL-TIME DEPTH ESTIMATION

Key Steps:

1. Decode the video stream in real-time.
2. Extract individual frames.
3. Detect pupils and extract coordinates.
4. Use these coordinates as inputs to the regression model to predict the corresponding gaze depth.

VIDEO CODING CONCEPTS

- Compression refers to the process of **reducing the amount of data** by compacting it into fewer bits.
- This process involves two complementary systems: a compressor (**encoder**) and a decompressor (**decoder**).
- Together, the encoder and decoder are referred to as CODEC.
- Tobii Pro 3 Glasses uses the H.264 standard.



Lossless

Retains all original data, ensuring that the decompressed video is identical to the original.

Lossy

Achieves higher compression ratios by discarding some data deemed less critical to the overall visual experience.

VIDEO ENCODER

Prediction Model

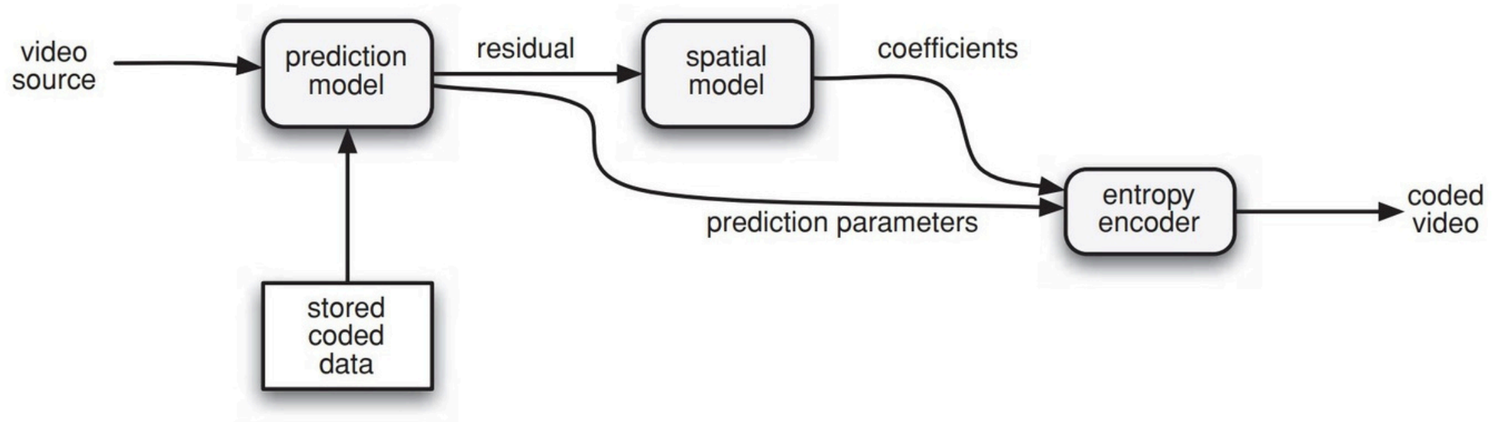
Reduce redundancy by exploiting similarities between neighboring video frames (temporal redundancy) and neighboring pixels within a frame (spatial redundancy).

Spatial Model

Works on the residual frame provided by the PM. Converts spatial domain data into frequency domain coefficients.
For further efficiency, coefficients are quantized.

Entropy Encoder

Compresses the parameters from the prediction model and the quantized transform coefficients from the spatial model.
This process removes statistical redundancy.



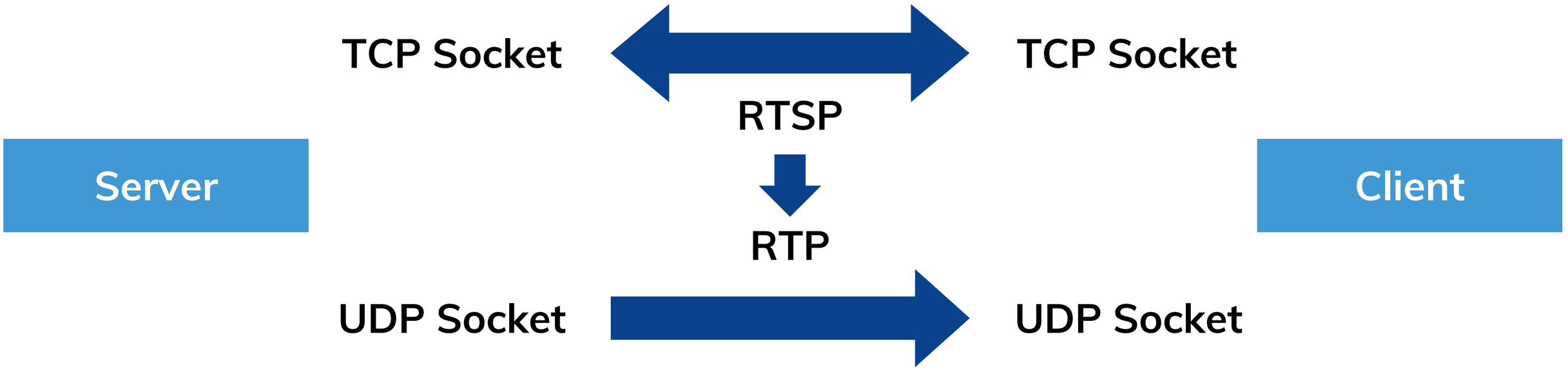
THE H.264 STANDARD

Prediction Model Features

- **Spatial Prediction**:
  - Frame divided in macroblocks (16x16 or 4x4)
  - Predict the pixel values of a block based on the pixel values of previously-coded blocks.
- **Temporal Prediction**:
  - exploits temporal redundancy by encoding differences between successive frames.
  - uses a range of block sizes (from 16x16 down to 4x4).

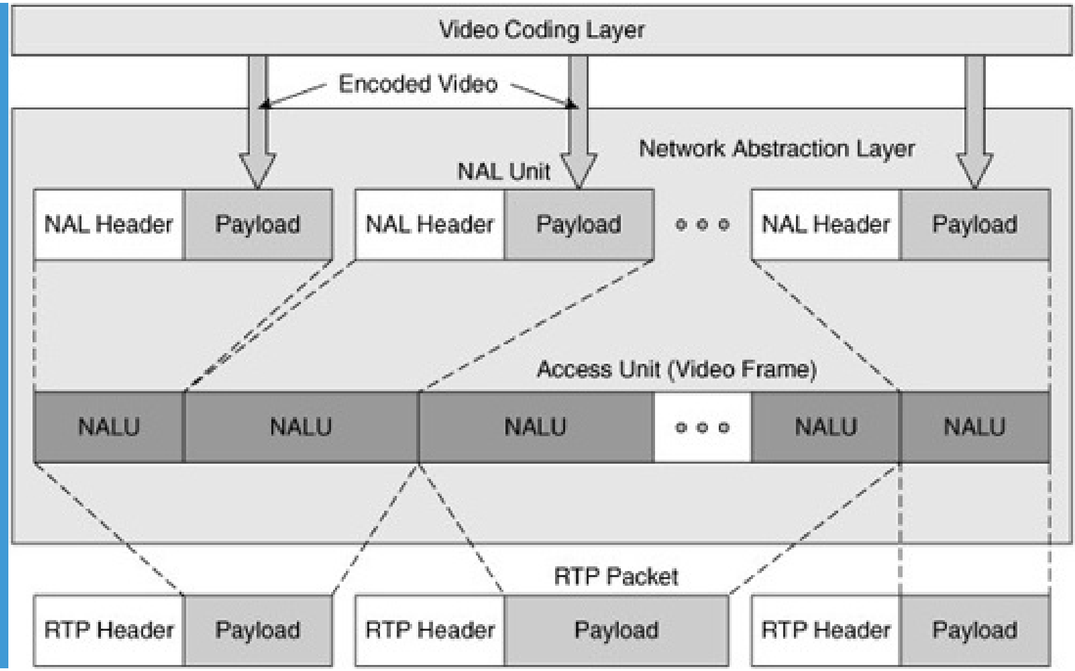
I	Intra-Coded Frames
P	Predictive-Coded Frames
B	Bi-Predictive-Coded Frames

REAL-TIME DECODING



REAL-TIME PROTOCOL PACKET

- RTP packets have a Header and Payload.
- Header:
  - 12 bytes long.
  - useful information about the packet itself.
- Payload:
  - contains one or more NAL units.
  - NAL units contain a portion of video data, like Sequence Parameter Set (SPS) or Picture Parameter Set (PPS).



REAL-TIME DECODING

RTP packets handling

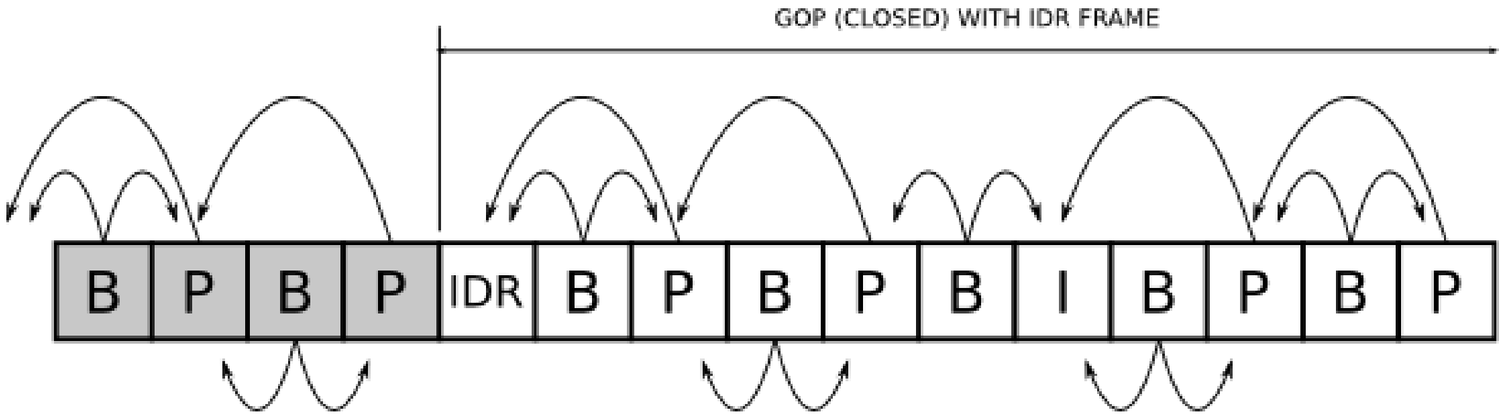
1. Extract NAL byte from the paylod and determine the type.
2. Handle different types such as PPS, SPS, FU.
3. Add start bytes for video decoders.

Group of Pictures

- Necessary to wait until a complete GOP is received.
- Each GOP starts with a SPS and a PPS.
- Following frames are collected until the next SPS and PPS are received.

Decode with ffmpeg

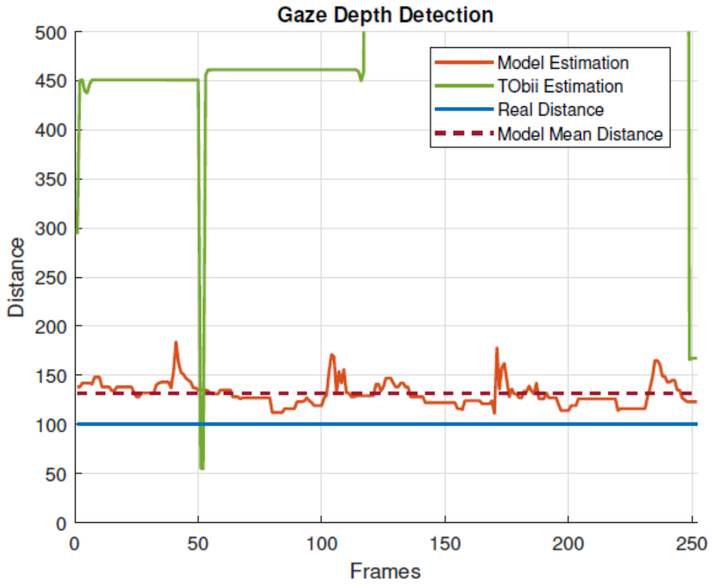
1. Feed GOP to the ffmpeg process to decode it.
2. Convert the output of the process to an array, which can be read by OpenCV as a frame.

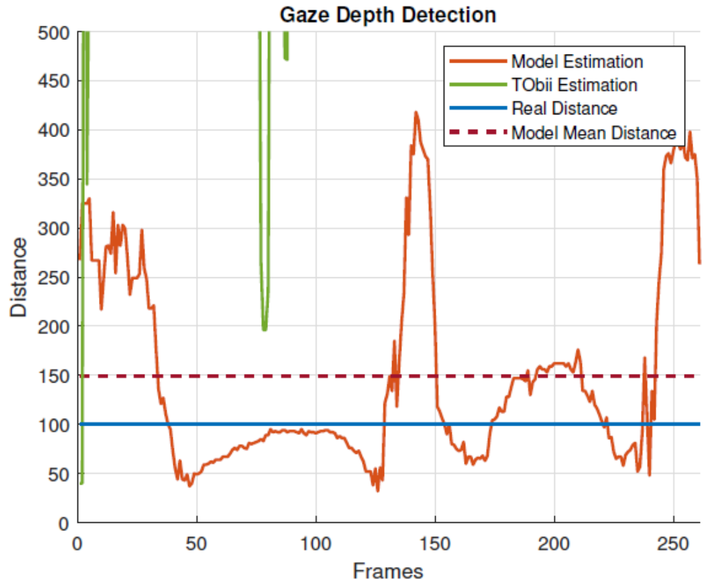


DEPTH ESTIMATION RESULTS

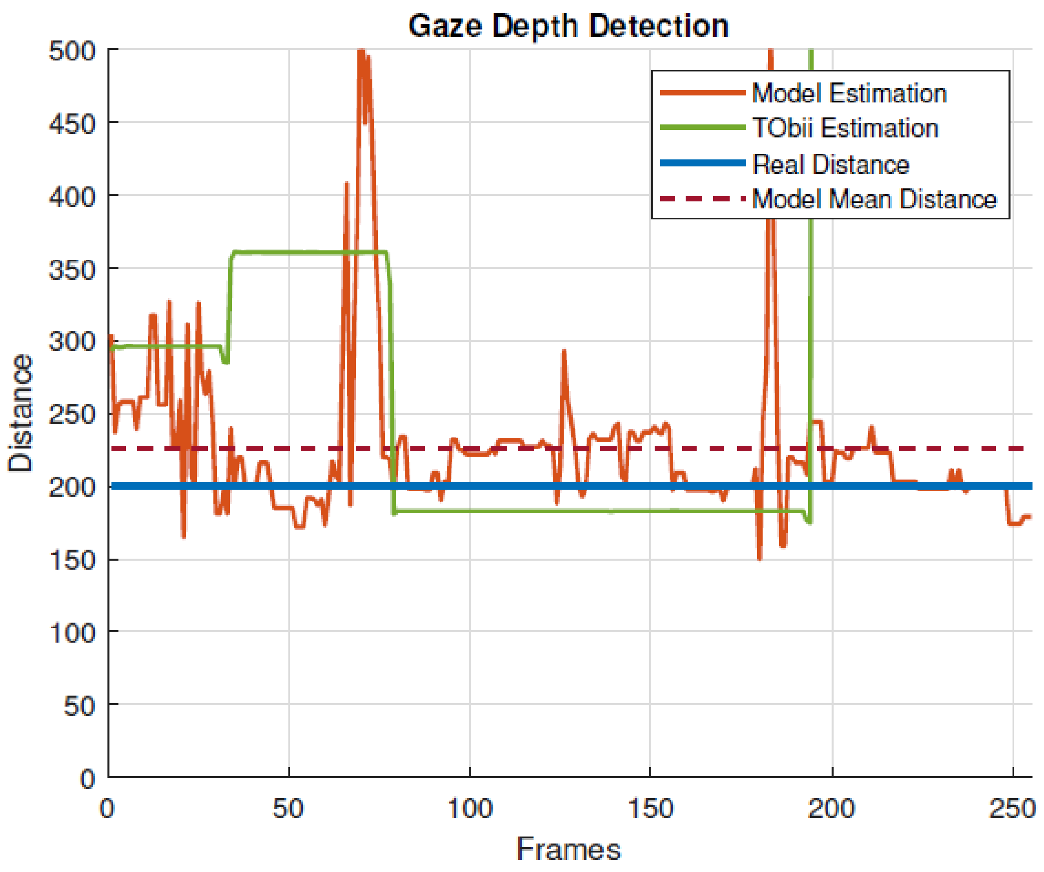
- Targets placed at 1,2, and 3 meters.
- Tested both frontally and for lateral targets
- Binarization method was used for pupil detection along with the regression model obtained with the second experiment.

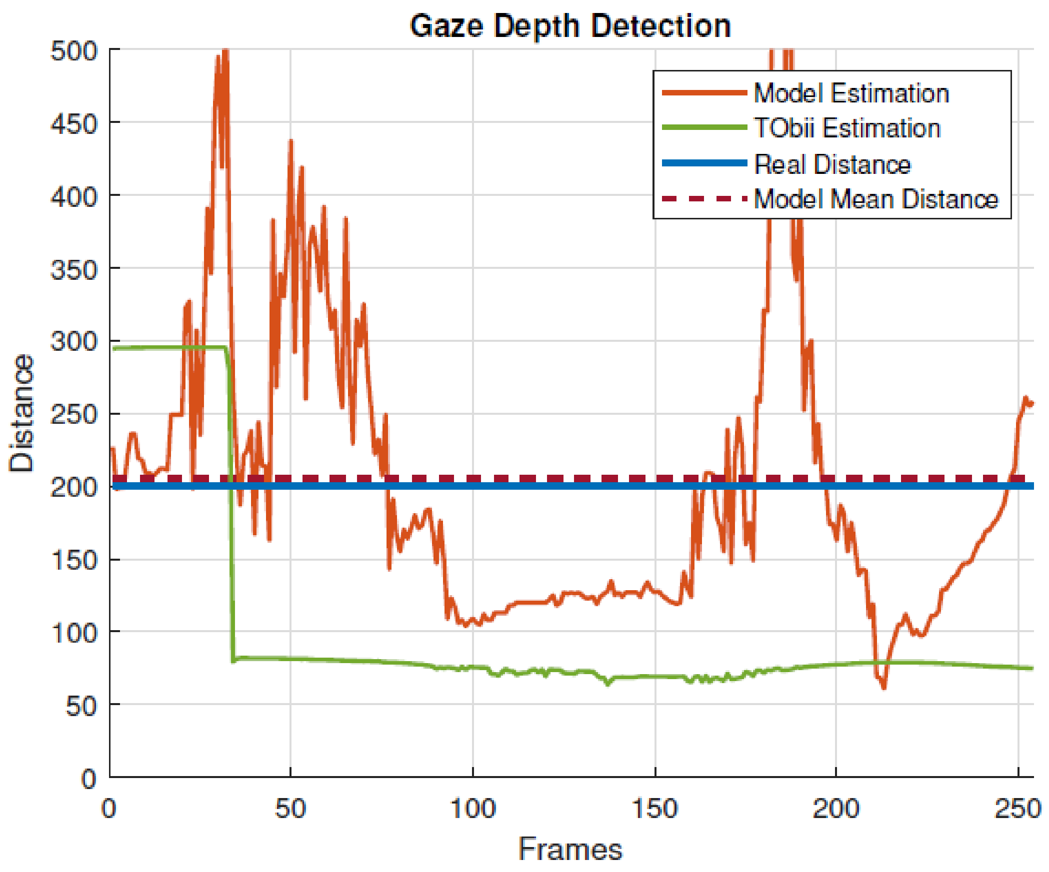
Mean Estimation	Frontal	Lateral
100 cm	131 cm	149 cm
200 cm	225 cm	205 cm
300 cm	271 cm	238 cm

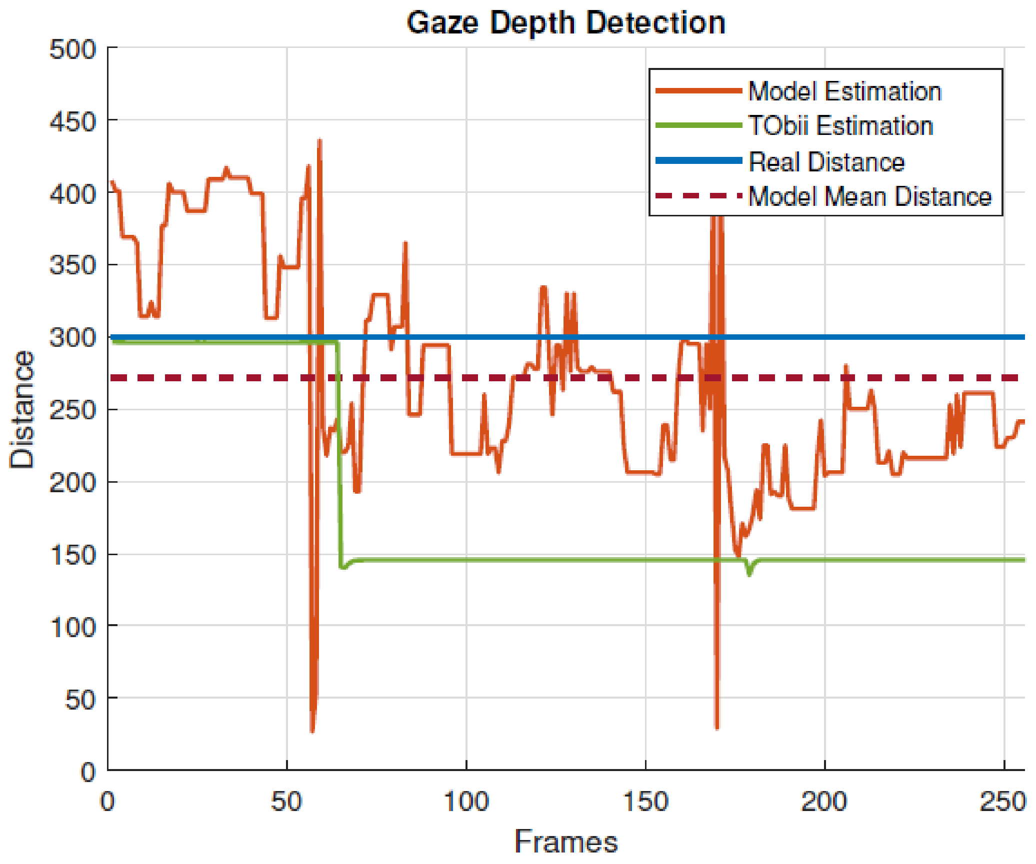


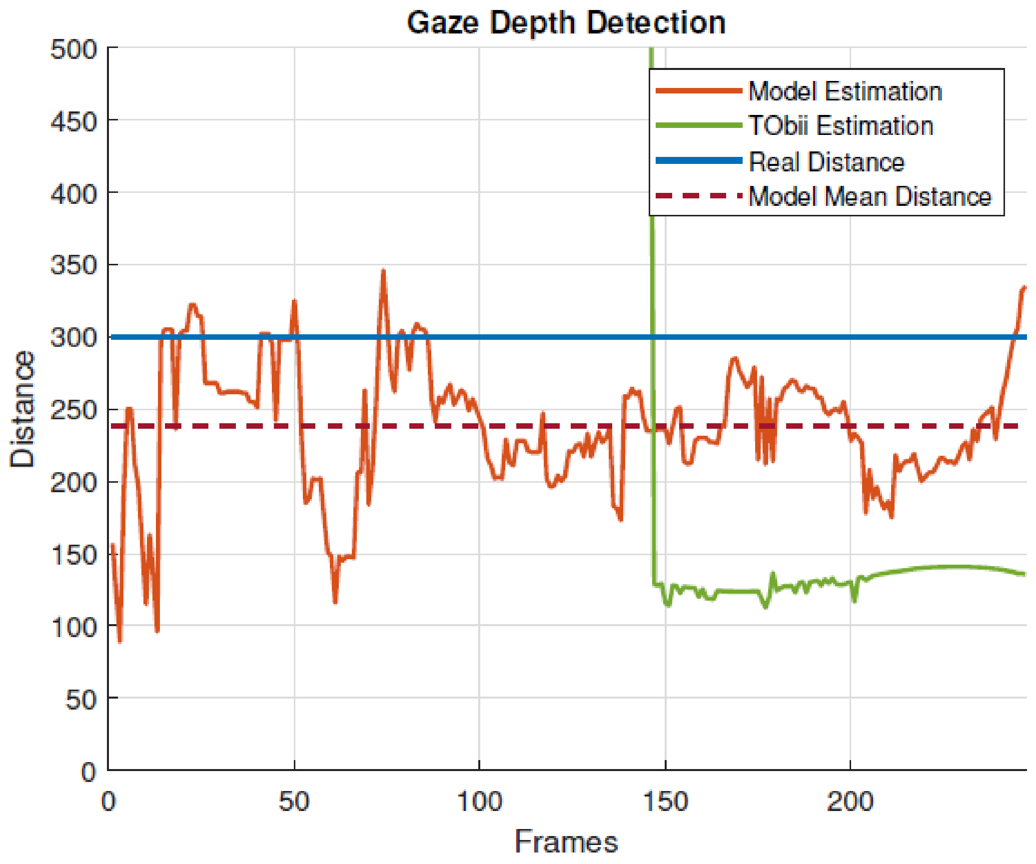


DEPTH ESTIMATION RESULTS



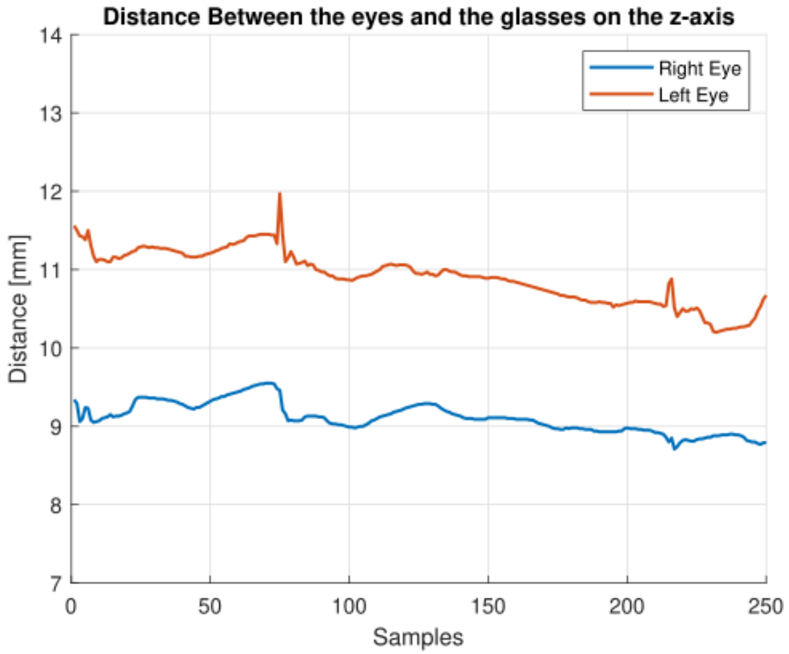






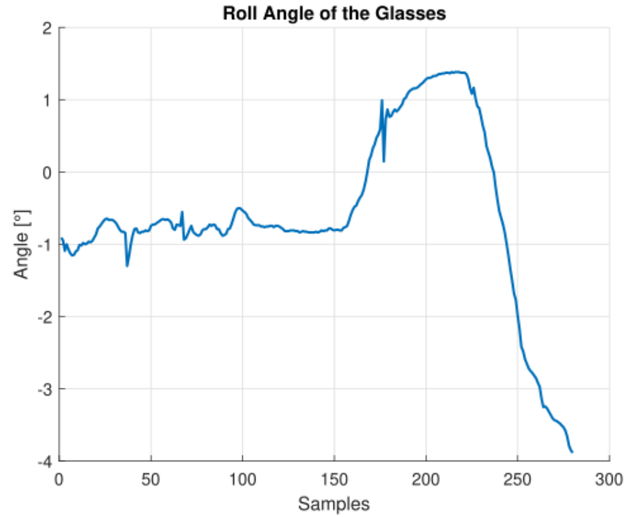
GLASSES' POSITIONING ESTIMATION

- The positioning of the glasses heavily impact depth estimation perfomance.
- Tobii provides his own estimation of pupil position wrt the frame of the glasses.
- The idea was to train a model that incorporates this additional information to improve the overall accuracy of depth estimation.
- Unfortunately data provided by Tobii regarding the z-axis of the pupil position was completely useless.

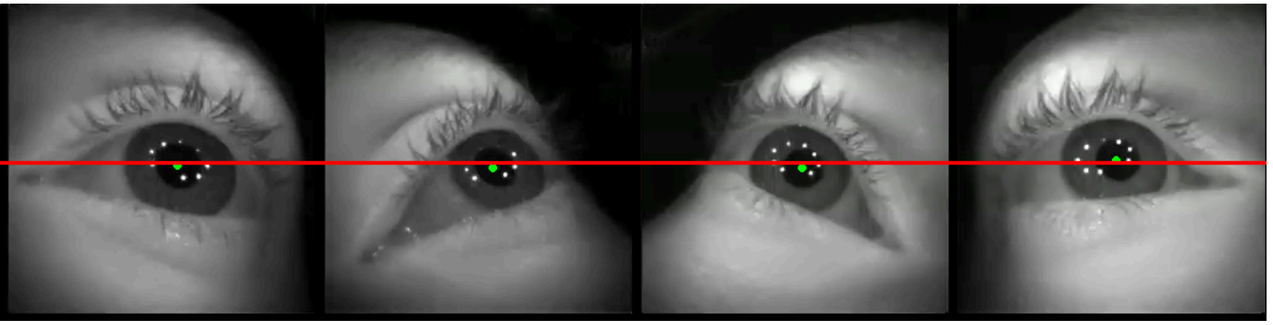


ROLL ANGLE ESTIMATION

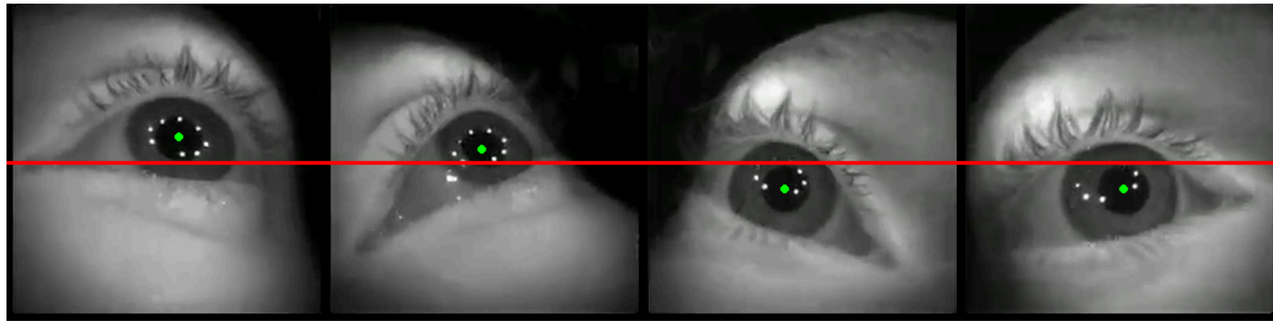
- Use Tobii's data to compute difference between the y-coordinates of the pupils.
- If the glasses are positioned correctly, the eyes should be at the same height.
- Knowing the distance between the pupils and their y-coordinates, the roll angle can be derived geometrically.



Straight



Tilted



CONCLUSIONS

1. Development of Pupil Detection procedures
2. Development of a Regression Model for Depth Estimation
3. Real-Time Decoding of the Video Stream from Tobii Pro Glasses 3
4. Glasses' Roll Angle Estimation

FUTURE WORK

1

Improving pupil detection algorithms to better handle variations in lighting and eye occlusions.

2

Expanding the dataset and exploring different machine learning techniques could refine the regression model further.

3

Integrating additional sensors, such as accelerometers or gyroscopes, may enhance the accuracy of glasses' positioning and depth estimation.

THANK YOU

Presented by Luca Secchieri

Universitat Politècnica de Catalunya