

ZHAW SCHOOL OF ENGINEERING

PROJEKTARBEIT

IT

Title

Authors:

Linda Helen BOEDI
Valentin BOSSI

Supervisor:

HANS-JOACHIM GELKE

November 27, 2017

Chapter 1

Acknowledgement

Chapter 2

Affidavit

Contents

1	Acknowledgement	1
2	Affidavit	2
3	Conceptual formulation	5
4	Motivation	6
5	Fundamentals	7
5.1	Competitor Analysis	7
5.1.1	Optical versus acoustical measurement	7
5.1.1.1	Acoustical measurement	7
5.1.1.2	Optical measurement	8
5.1.2	Companies	8
5.1.2.1	Witschi - WisioScope S	8
5.1.2.2	Lepsi - WatchScope/WatchAnalyzer	8
5.1.2.3	Greiner Vibrograf - Compact 900	9
5.2	Pi camera module	9
5.2.1	Exposure time	11
5.2.1.1	Minimal exposure time	11
5.2.1.2	Maximum framerate	11
5.2.1.3	Maximum exposure time	12
5.2.2	Hardware limits	12
5.3	Reference measurement and test watch	12
5.3.1	Number of impacts	12
6	Experimental approaches	14
6.1	Evaluation of possible technologies	14
6.1.1	Frame analysis versus given GPU motion vectors	14
6.1.2	Frame analysis via OpenCV	14
6.1.3	Motion vectors from high level library picamera	15

6.2	Animation of motion vectors	15
6.2.1	Motion vectors visualized in a frame	17
6.3	Picamera library	19
6.3.1	Picamera settings	19
6.4	Crude calculation of frequency	19
6.5	Verification of accuracy	21
7	Calculation of the frequency	22
7.1	Implementation of the algorithm	22
8	Results	23
8.1	Measurements	23
8.2	Calculation of gang deviation from frequency	24
8.3	Evaluation	24
	Glossary	25

Chapter 3

Conceptual formulation

Chapter 4

Motivation

Chapter 5

Fundamentals

5.1 Competitor Analysis

5.1.1 Optical versus acoustical measurement

To measure the frequency of the balance wheel of a mechanic clock there exist mainly two methods on the market, acoustical and optical measurement. The common used method is by analyzing the noises of the balance wheel. More expensive devices use both acoustical and optical feedback. In order to be able to classify the rate deviation well enough, it is usually measured in different positions of the watch.

5.1.1.1 Acoustical measurement

The first used method to get the frequency measurement of the balance wheel was by using a vibrograph was used. For every "tick" of the clock a line was drawn onto a ongoing strip of paper. Out of the distance between of these lines the frequency was calculated. (1) But as this method isn't the most accurate other techniques are used nowadays. Modern watch timing

Comparison acoustical and optical measurement		
	acoustical	optical
since	around 19th century (2)	end of 20th century (2)
accuracy	0.1 s/d	0.1 s/d
advantages	most experience (exists for about 200 years)	measurement can be done anywhere
disadvantages	background noises need to be filtered out	

machines use an oscillating quartz crystal as comparison for the frequency of the balance wheel (1). The noises of the balance wheel are recorded and amplified with a microphone whereat unwanted background noises need to be filtered out.

5.1.1.2 Optical measurement

About hundred years later, optical watch timing machines joined the acoustical ones. There are barely devices on the market, which use only optical measurement, but several companies started to combine their acoustical with optical metering. One way to scale the frequency of the balance wheel optically is by using a laser. The beam will be periodically interrupted by the balance wheel and thus the frequency can be calculated (3). In this paper the frequency should be quantified optically by using image processing. There is hardly to none company out there, which uses the last technique for measuring the frequency.

5.1.2 Companies

5.1.2.1 Witschi - WisioScope S

WisioScope S tests mechanical watches acoustically and optically. The measurement is done parallel and in this way is more accurate as both signals are used for the calculation of the frequency. The optical metering is done using a laser and lighting, a camera helps to adjust the watch properly. The costs of their product are: CHF 10450.-

Measurement gear deviation	
scope	+/- 999.9 s/d
resolution	0.1 s/d
price	CHF 10450.-

5.1.2.2 Lepsi - WatchScope/WatchAnalyzer

The WatchScope and WatchAnalyzer from Lepsi are especially for lovers and not really for production. Both products are Swiss technologies. Both devices only work with an acoustic input, with measurements of either a few seconds or up to 24 hours. All data can then be queried with the smartphone. WatchScope is the smaller, more convenient version. With the WatchAnalyzer you can easily measure the rate deviation of the watch in several positions. The prices for this product are CHF 369.00, resp. CHF 929.-

Measurement gear deviation	
scope	+/- 1000 s/d
resolution	0.1 s/d
price	CHF 369.-, resp. CHF 929.-

5.1.2.3 Greiner Vibrograf - Compact 900

The Compact 900 measures only the beat noises of the balance wheel. The price is CHF 4070.-

Measurement gear deviation	
scope	+/- 1000 s/d
resolution	0.1 s/d
price	CHF 4070.-

5.2 Pi camera module

The camera module of the Pi is basically a mobile phone camera module. Among other things, it uses a rolling shutter to take pictures. The pixel values of a frame are not captured completely at once, but are read line by line.

However, the sensor is configured via the registers with the number of lines to be read and a corresponding time. The sensor reads the lines and pushes the data with the configured speed to the Raspberry PIPi.

This keeps the readout time of each line constant. However, the CPU does not speak directly to the camera, but processing takes place on the GPU (VideoCore IV) of the Raspberry Pi, which operates its own real-time operating system (VCOS).

The illustration above illustrates the processing flow of a frame, with the associated steps explained in the following section.

1. The camera's sensor is configured and continuously streams frame lines to the GPU.
2. The GPU builds complete frame buffers from these lines and performs the post-processing of these buffers.

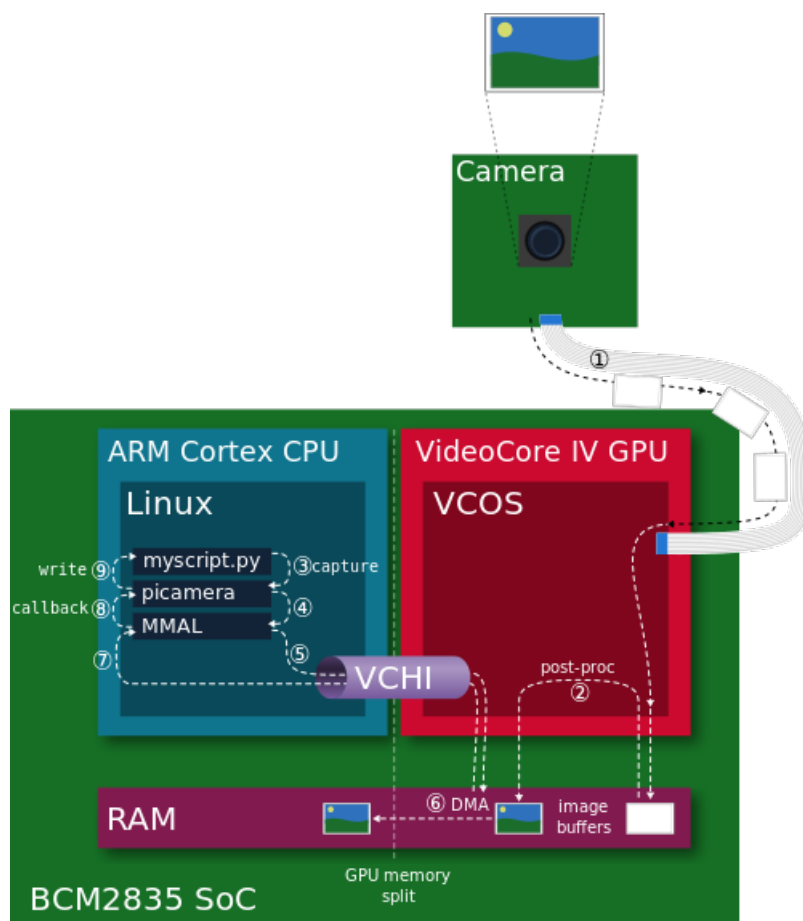


Figure 5.1: Camera architecture. Readthedocs Picamera

3. Meanwhile, `myscript.py` makes a capture call with the `picamera` on the CPU.
4. The `Picamera` library uses the MMAL API to meet this requirement.
5. The MMAL API sends a message via VCHI requesting a frame capture.
6. The GPU then initiates a DMA transmission of the next full frame from its RAM portion to the CPU portion.
7. Finally, the GPU sends a message via VCHI that the capture is complete.
8. This causes an MMAL thread to trigger a callback in the `Picamera` library, which in turn retrieves the frame.
9. Finally, `picamera` calls to the output object provided by `myscript.py`.

5.2.1 Exposure time

The camera sensor detects how many photons hit the sensor elements, because the more impact, the more they increase their counter values. The sensor can perform exactly two operations; reset a set of elements or read a set of elements.

5.2.1.1 Minimal exposure time

Reading out a series of elements takes a certain amount of time, thus there is a limit to the minimum exposure time. Assuming one has 500 lines on a sensor and reading each line takes at least 20ns, then it will take at least $500 * 20\text{ns} = 10\text{ms}$ to read a full screen.

5.2.1.2 Maximum framerate

The frame rate is the number of frames the camera can capture per second. The exposure time determines the maximum number of images that can be taken in a given time. Assuming it takes 10ms to read a complete image, then no more than $\frac{1\text{s}}{10\text{ms}} = \frac{1\text{s}}{0.01\text{s}} = 100$ images in one second.

So it is valid:

$$\frac{1\text{s}}{\text{min exposure time in s}} = \text{max framerate in fps.}$$

The lower the minimum exposure time, the higher the maximum frame rate and vice versa.

5.2.1.3 Maximum exposure time

Um die Belichtungszeit zu maximieren, muss die Framerate reduziert werden.

Es gilt :

$$\frac{1\text{s}}{\text{min framerate in fps}} = \text{max exposure time in s}$$

5.2.2 Hardware limits

- The maximum resolution for MJPEG recording is partially dependent on the GPU memory.
- The maximum horizontal resolution for the standard H264 recording is 1920 (this is a limitation of the H264 block in the GPU).
- The maximum frame rate of the camera depends on several factors. With overclocking 120fps can be achieved, but 90fps is the maximum supported frame rate.

5.3 Reference measurement and test watch

At first some general information to the test watch, which was used.

The test watch is a watch of the caliber ETA C01.211 it is a chronograph, with a diameter of 31 centimeters. It has a beat rate of 21600 beats per hour and therefore a frequency of 3 Hz. Additionally the watch has a power reserve of 43 hours. (5)

In order to obtain an exact reference value of the frequency of the balance wheel of the test watch, an acoustic measurement was carried out with a professional measuring device, the Greiner Vibrograf - Compact 900, and the rate deviation was determined.

In order to calculate the frequency from the gear deviation of the balance wheel, first some transformations had to be carried out.

5.3.1 Number of impacts

Since the rate deviation has been calculated in seconds per day and the number of strokes per hour, i. e. the number of audible impulses (beats) of a two-armed armature of a mechanical watch per hour, is known, these two

Number of impacts [1/h]	Impact duration [s]	Oscillation number [1/h]	Period duration [s]	Frequency [Hz]
18'000	0.200	9'000	0.400	2.50
19'800	0.182	9'900	0.364	2.74
21'600	0.166	10'800	0.333	3.00
28'800	0.125	14'400	0.250	4.00
36'000	0.100	18'000	0.200	5.00

Illustration 1: Number of strokes, period duration and frequency of the balance of automatic wristwatches

quantities must first be brought to a unit. The relationship between the beat number (n^*) and frequency (f) is as follows:

$$n^* = 2f * 3600$$

and therefore:

$$f = \frac{n^*}{7200}$$

Chapter 6

Experimental approaches

6.1 Evaluation of possible technologies

6.1.1 Frame analysis versus given GPU motion vectors

One approach is to compare each image with the subsequent image and to mark the displacements with motion vectors. This would mean a considerable computational effort. Such motion vectors are already made by the GPU, in order to compress videos in the H.264 codec.

6.1.2 Frame analysis via OpenCV

With the library OpenCV it is possible to determine the optical flow of an image sequence.

The optical flow represents the vector field of the velocities of the points of an image sequence. It is a useful representation of motion information in image processing. The local optical flow is an estimate of patterns in an image in the vicinity of a viewed pixel.

One possibility would be to follow the course of a velocity vector and thereby calculate a period of the frequency of the balance wheel. The calculation of the flow at selected points is also called feature point tracking. Another approach would be to calculate the frequency based on the velocity obtained from the optical flow, since it is more or less a circular movement.

One difficulty with this procedure is that one has to commit to certain pixels that one examines. This can cause problems if the clock is placed differently, because wrong points (worst-case scenario: points in which movement never takes place) can cause problems for the calculation.

The OpenCV library uses the Lucas-Kanade method to calculate the optical flow. The Lucas-Kanade method also assumes that the flow in the

local environment of the pixel for which the flow is intended is the same. So more or less a whole set of pixels is considered. The flow can thus be determined by calculating the derivatives (= gradients).

One characteristic of this method is that it does not provide a dense flow, i. e. the flow information disappears quickly with the distance from the edges of the image. The advantage of the method is its relative robustness against noise and smaller defects in the image.

6.1.3 Motion vectors from high level library picamera

The picamera library is capable of storing all motion vector estimations of the h.264 encoder in an object. This object with all motion vectors for all frames can be later accessed and used. The motion data is at the level of macro-blocks where the values are 4-bytes long and consist of a signed 1-byte x vector, a signed 1-byte y vector and an unsigned 2-byte SAD (Sum of Absolute Differences) value. All motion data values can be easily accessed frame by frame. (2)

The sum of absolute differences is a positive number resulting from the formation of the difference between two digital images. It serves as a measure of the difference between two images.

The SAD is obtained by subtracting the color values of the images pixel by pixel from each other and adding them up by amount. (3)

The advantage of the Picamera library is that compared to the possibilities of the OpenCV library, the information about the motion vectors is available faster, since this library already has an internal object (motion-data), which stores all necessary information as well as the SAD value.

6.2 Animation of motion vectors

After deciding which library to use, some experimental sessions took place to get a better understanding of the so called motion data values or motion vectors. With the picamera library it is easy to access these motion data values so the hard part wasn't to get the values but to understand how they are constructed. The motion data object consists of multiple values; a signed 1-byte x vector, a signed 1-byte y vector and an unsigned 2-byte SAD value for each macro-block of a frame.

In a first step all SAD values were animated with colors to get a better understanding of how good the information they keep is and how as well as if they can be used to calculate the frequency of the movement of the balance wheel.

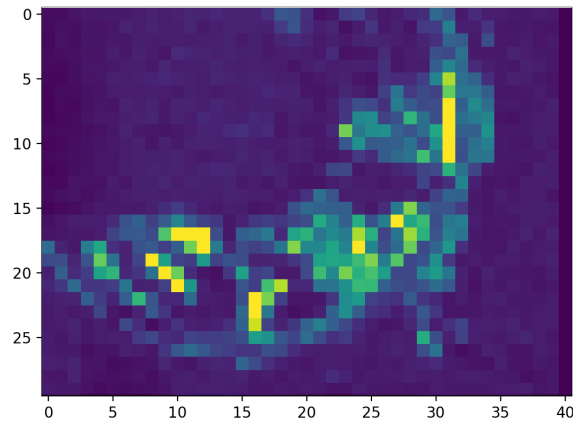


Figure 6.1: Animation of the SAD values, the lighter the color the higher the value of the SAD

Further the x and y vector were analyzed and also animated. But as those vectors itself are hard to use and give incomplete information (WHY?), the hypotenuses were calculated with Pythagoras' theorem and displayed similar to the SAD values. The hypotenuses give information about the amount as well as the direction of the motion.

Another experimental approach was displaying the motion vectors as arrows directly in the video. This approach was helpful to get a better understanding under which conditions the best result of the motion vectors is reached.

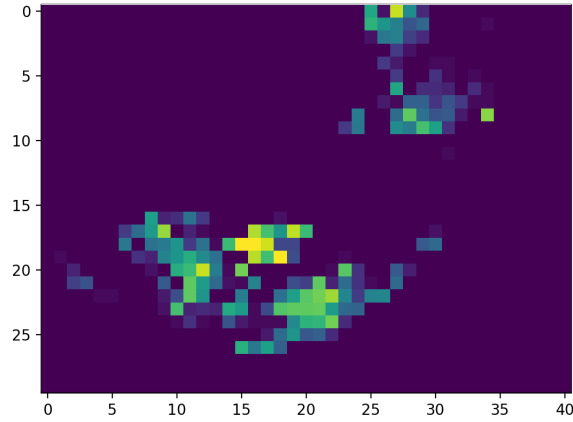


Figure 6.2: Animation of the hypotenuses, the lighter the color the higher the value of the hypotenuse

6.2.1 Motion vectors visualized in a frame

Codecvisa was used to display the motion vectors of a frame. The result is shown in Figure 3.

Another approach used Excel was to determine whether a motion vector from one frame is the predecessor of the motion vector in the next frame. The hope was that a moving pixel could be followed and therefore the location of the pixel would always have been known. The result was sobering because one frame and motion vector cannot be associated with another frame and motion vector. This experiment looked at the idea of the optical flux with the movement vectors of the MMAL (Multi-Media Abstraction Layer).



Figure 6.3: Motion vectors drawn in one frame

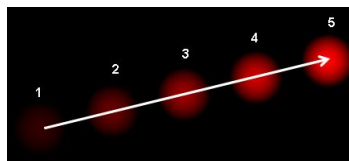


Figure 6.4: It shows a ball moving in 5 consecutive frames. The arrow shows its displacement vector.

6.3 Picamera library

The Picamera library is an open source project in the programming language Python. The library is an interface to the MMAL —written in C— which in turn is an interface to the firmware of the GPU. This means that the camera can be accessed using a simple programming language such as Python.

6.3.1 Picamera settings

Testing with different settings has shown that the results can be improved. For example, the black and white shots are the better choice. Reducing the shutter speeds can reduce motion blur, which also contributes to a better result. The consequence of the shorter exposure time is that the light source must be very strong and close to the subject, otherwise the image is too dark and has a negative effect on the result.

6.4 Crude calculation of frequency

With the help of the motion vectors and the corresponding time stamps per frame, a rough calculation of the frequency can be made. All x, y or SAD values contained in the image are summed up. If the sum of all values is small, it is very likely that the balance wheel is at a standstill before turning back again. For small values, the time stamp is registered and the difference to the next standstill is calculated. Since the balance wheel triggers half a oscillation, it has moved from one stop to the next half point. In order to roughly calculate the frequency, an Excel sheet was created in which all the accumulated x- and y-values were entered and the standstills of the balance wheels were determined by eye. The half-oscillations and the frequency were calculated from the time intervals of the standstills. Afterwards, the average of all calculated frequencies was determined and it was recognized that the results with this method are good enough to determine the frequency and thus the gear deviation.

xSumAbs	ySumAbs	timestamp		timestamp	stillstand	half period	period	hertz	hertz average
0	0	None		88645	166210	-			3.032792904
3050	2704	11081		254855	166209	-	332419	3.00825163	
2540	1770	22161		421064	177290	-			
1106	1464	33242		598954	155129	-	332419	3.00825163	
3422	2870	44324		753483	177289	-			
3070	2352	55403		930772	166210	-	343499	2.91121663	
1924	1658	66484		1096982	166210	-			
1012	1438	77565		1263192	166209	-	332419	3.00825163	
66	146	88645	88645	1429401	166209	-			
500	982	99726		1595610	166210	-	332419	3.00825163	
21266	20532	110807		1761820	166209	-			
2860	2780	121887		1928029	166210	-	332419	3.00825163	
3202	2546	132967		2094239	166209	-			
3322	2986	144048		2260448	166209	-	332418	3.00826068	
3342	2412	155129		2426657	166210	-			
2164	2730	166210		2592867	166209	-	332419	3.00825163	
3192	3012	177291		2759076	166209	-			
3142	2736	188371		2925285	166210	-	332419	3.00825163	
2236	2146	199452		3091495	166209	-			
2392	3152	210533		3257704	166209	-	332418	3.00826068	
3108	3434	221613		3423913	166210	-			
3480	2130	232693		3590123	166209	-	332419	3.00825163	
2204	1290	243775		3756332	166211	-			
354	440	254855	254855	3922543	166208	-	332419	3.00825163	
754	496	265935		4088751	166209	-			
3160	2034	277016		4254960	177292	-	343501	2.91119968	
3156	2560	288097		4432252	166208	-			
3264	3118	299178		4598460	166209	-	332417	3.00826973	
3426	2736	310258		4764669	166209	-			
2282	2934	321138		4930878	166210	-	332419	3.00825163	
3046	3390	332419		5097088	166209	-			
3682	4380	343500		5263297	166210	-	332419	3.00825163	
3168	2472	354580		5429507	166210	-			
2294	2394	365661		5595717	166208	-	332418	3.00826068	
3178	3328	376742		5761925	166210	-			
3354	2864	387822		5928135	166210	-	332420	3.00824258	
2708	2438	398903		6094345	166209	-			
1394	1604	409984		6260554	166209	-	332418	3.00826068	
192	456	421064	421064	6426763	166210	-			
516	908	432145		6592973	166209	-	332419	3.00825163	
1724	1894	443226		6759182	166210	-			
3006	2476	454306		6925392	166209	-	332419	3.00825163	
3732	2800	465387		7091601	166209	-			
3586	3248	476467		7257810	166211	-	332420	3.00824258	
3152	2598	487548		7424021	166208	-			

Figure 6.5: Excel sheet with summed x-, y-values, timestamps and calculated frequency

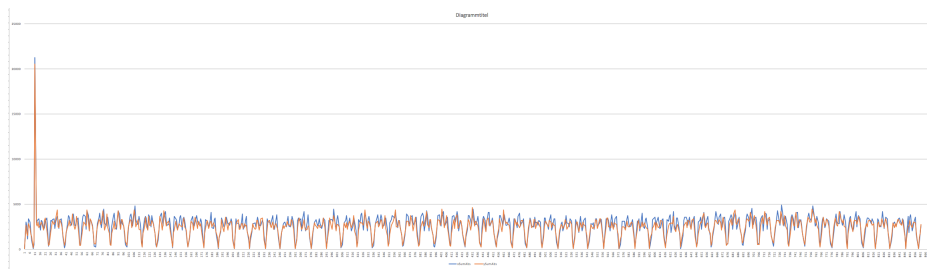


Figure 6.6: Illustration of summed x-, y-values over time

6.5 Verification of accuracy

The calculated frequency is far too imprecise. In theory, however, the calculation should become more precise after a certain period of time and should even be accurate to the microsecond. But this has never been achieved. For this reason, a reliable artificial clock was used in the form of an LED that is clocked by a 50 Mhz quartz and is therefore sufficiently accurate.

Duration	run 1	run 2	run 3	run 4	run 5
10 sec	166591.5	166585.0	166397.2	165655.4	166585.0
20 sec	166679.0	166117.1	166675.0	166675.0	166581.9
40 sec	166486.4	166673.1	166675.0	166440.3	166673.1
60 sec	166673.7	166672.4	166673.7	166672.4	166673.7
300 sec	166665.2	166671.7	166640.4	166665.3	166640.4
600 sec	166668.2	166649.6	166668.2	166655.8	166668.2
900 sec	166667.2	166667.3	166638.4	166638.4	166667.2
1200 sec	166660.4	166668.3	166669.7	166668.2	166668.2
1500 sec	166668.8	166650.3	166650.3	166668.8	166668.8
1800 sec	166652.8	166668.2	166669.2	166668.2	166669.2
2100 sec	166668.6	166668.6	166668.6	166655.4	166654.5
2400 sec	166668.1	166652.7	166656.5	166668.1	166668.1

Chapter 7

Calculation of the frequency

7.1 Implementation of the algorithm

In a next step, the results of the rough calculation from the Excel sheet were transferred into a Python program.

The greatest difficulty proved to be the detection of the minima, which characterize the standstill of the balance wheel. These were relatively easy visible to the unaided eye, but in an automatic calculation, appropriate conditions had to be set in order to take into account the correct values and correct the inaccuracies caused by the noise. The setting of an upper limit for the minima proved to be a suitable condition. All minima below this limit are therefore taken into account.

Furthermore, the calculation of the frequency in the Python program has been further simplified by not calculating each period individually, but by selecting the first and the last minimum over the whole period of time, in which it was recorded. This time span is then calculated by half of the number of minima minus 1. Half, because every half of the period the balance wheel stops and minus 1, because otherwise a half-period would be taken into account too much. This approach reduces the number of rounding errors, resulting in higher accuracy. After this calculation, the period duration T is now available, from which the frequency f can be calculated quite simply:

$$f = \frac{1}{T}$$

Chapter 8

Results

8.1 Measurements

This measurements were taken in different positions of the watch with different length of recordings. Each position were recorded in 10, 20, 40 and 80 minutes.

Figure 5 and Table 3 shows the deviation per day based on the following calculation: Taken the first timestamp (t_1) of a stillstand and taken the last timestamp (t_2) of a stillstand, the counted number of stillstands (s_1), number of strokes per hour (s_2) (21600):

$$\begin{aligned} t_2 - t_1 &= \Delta t \\ \Delta t / s &= f * 2 \\ (f * 2 * s_2) - 3600 &= \text{deviation per hour} \end{aligned}$$

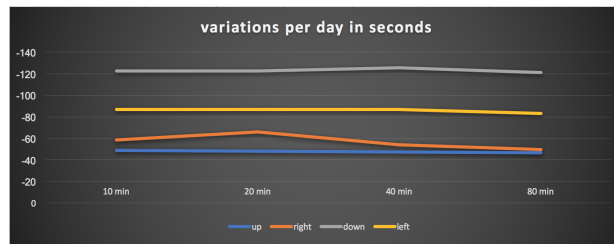


Figure 8.1: Variation of different time and position recordings

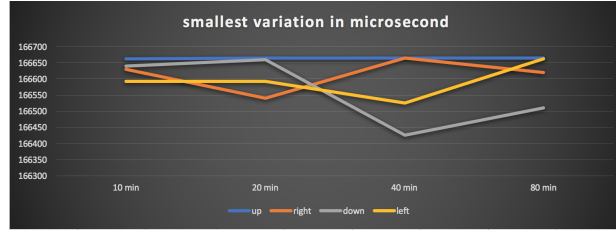


Figure 8.2: Variation of different time and position recordings

Positions	Greiner Compact 900	10 min	20 min	40 min	80 min
up	-13	-48.7	-48.0	-47.6	-47.0
right	-30	-58.3	-65.5	-54.4	-49.4
down	-20	-122.2	-122.2	-125.4	-121.4
left	-14	-87.0	-87.1	-87.1	-83.1

Table 3: variations per day in seconds

Positions	Wisio Scope	10 min	20 min	40 min	80 min
up	-32	-48.7	-48.0	-47.6	-47.0
right	-60	-58.3	-65.5	-54.4	-49.4
down	-43	-122.2	-122.2	-125.4	-121.4
left	-29	-87.0	-87.1	-87.1	-83.1

Table 4: variations per day in seconds based on smallest variation

8.2 Calculation of gang deviation from frequency

8.3 Evaluation

Glossary

Balance wheel is a wheel that regulates or stabilizes the motion of a mechanical watch and is responsible for the clock to run precisely. 20

MMAL (Multimedia Abstraction Layer) is a framework designed by Broadcom to provide a host-side, simple and relatively low-level interface to multimedia components running on the Videocore IV GPU on the Raspberry Pi. 20

Optical flow The optical flow of an image sequence is the vector field of the velocity projected into the image from visible points of the object in the reference system of the imaging optics. 20

Rolling shutter When the camera needs to capture an image, it reads out pixels from the sensor a row at a time rather than capturing all pixel values at once. 20

SAD (sum of absolute differences) is a positive number, which results from the formation of the difference between two digital images. It serves as a measure of the difference between two images and is used in image processing and pattern recognition. It is obtained by subtracting the color values of the images pixel by pixel from each other and adding them up by amount. 20

Bibliography

[Uhrenwiki] (1) <https://www.uhren-wiki.net/index.php?title=Zeitwaage>

[ncsli] <https://www.ncsli.org/c/f/p11/286.314.pdf>

[chronoskop] <http://www.chronoskop.com/index.php/chronoskop-zeitwaagen-von-preislisit-timegraphers/>

[Picamera] 2) picamera readthedocs

[SAD] 3) Wikipedia: Sum of absolute differences

[Schlagzahl] 4) Wikipedia: Schlagzahl

[Caliber Test watch] 5) <http://watchbase.com/eta/caliber/c01-211>

List of Figures

5.1	Camera architecture. Readthedocs Picamera	10
6.1	Animation of the SAD values, the lighter the color the higher the value of the SAD	16
6.2	Animation of the hypotenuses, the lighter the color the higher the value of the hypotenuse	17
6.3	Motion vectors drawn in one frame	18
6.4	It shows a ball moving in 5 consecutive frames. The arrow shows its displacement vector.	18
6.5	Excel sheet with summed x-, y-values, timestamps and calcu- lated frequency	20
6.6	Illustration of summed x-, y-values over time	20
8.1	Variation of different time and position recordings	23
8.2	Variation of different time and position recordings	24