Effects of light intensity on motility three species of copepods: *Acartia tonsa, Calanus finmarchicus* and *Temora longicornis*

Pietro Antonio Grassi

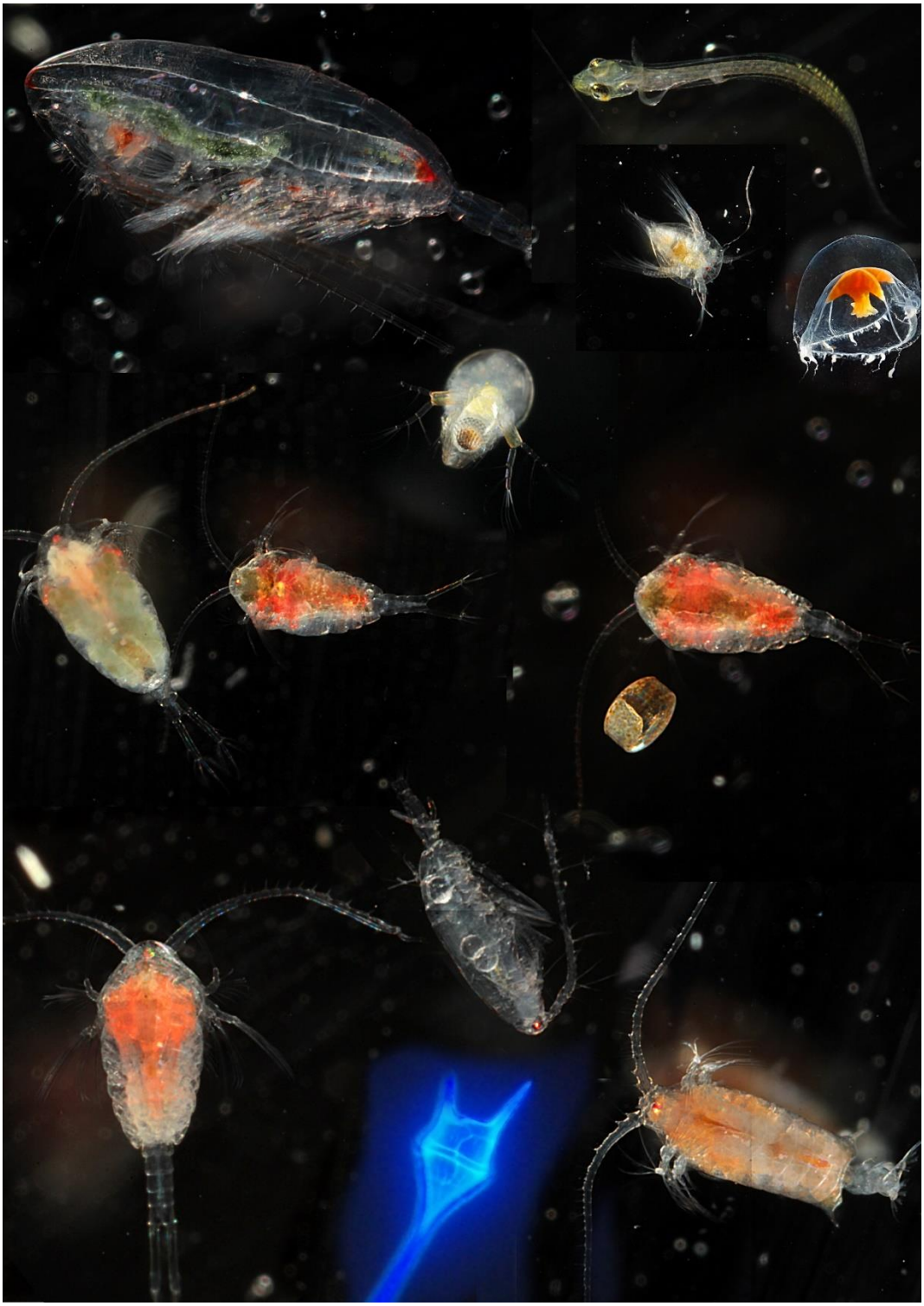
Master thesis

Matematisk-Naturvitenskapelige Fakultet

UNIVERSITETET I OSLO

01.12.2017

Effects of light intensity on motility three species of copepods: *Acartia tonsa, Calanus finmarchicus* and *Temora longicornis*



År 2017

Effects of light intensity on motility three species of copepods: Acartia tonsa, Calanus finmarchicus and Temora longicornis

Pietro Antonio Grassi

http://www.duo.uio.no/

Trykk: Reprosentralen, Universitetet i Oslo

Abstract

Light is one of the most important structuring factors for life in the ocean. For zooplankton grazers, including copepods, light offers a behavioral dilemma that is potentially present at both small and large scales. Copepods rely on phytoplankton that in turn rely on light for photosynthesis. In the same time copepods need to avoid being detected by visually feeding fish that see better in light environments. In the first part of my project I designed and built a setup where light could be carefully manipulated and copepod behavior could be video filmed and automatically tracked. In the second part of my project I investigated how variation in light intensity influence motility of three common copepods (*Acartia clausi, Temora longicornins* and *Calanus finmarchicus*). The copepods were chosen because they vary in size and color, both of which affect detection by visual predators. Copepods were exposed to random sequences of 5 light intensities separated by periods of darkness. Light intensity ranged between 0.06 and 2.67 µE $m^{-2}s^{-1}$ with a LED NeoPixel as source light. From the video analysis I obtained data of the position, and hence speed, for each animal that was subsequently analyzed statistically. The copepods responded to the light stimuli with different motility reactions. *Temora* and *Calanus* had a negative phototactic behavior, while, *Acartia* had a positive one. The light intensity per se did not have a strong influence on distance from the light source in terms of directionality. While swimming speed was similar at all light intensities for *Calanus* and *Temora*, *Acartia* swam faster at the two lowest light intenties. All in all my results suggests that both size and coloration affect motility behavior in response to light.My setup allows for easy manipulating the spectral composition and the intensity of the light. it is thus particularly suited to address the above questions, either with focus on light or in combination with other environmental drivers.

Acknowledgements

I want to write a few simple words. Thank you to Josefin Titelman, Tom Andersen, and Jan Heuschele, my supervisors, and mentors in this adventure at UiO. Thank you for the friendly and pleasant environment where I spent these two years. To have shared your knowledge with me, and the help with the thesis. A special thank you to Jan that helped me thousand times with R and Python.

Special thanks to my wife, and future mother of my child, that support me and tolerates every time there is an exam. My family that supports me, although they are thousands of kilometers far from us.

Thanks to all student that had shared with me the experiences and the funny moments during the filed course in Drøbak and the laboratory activities. All professors and employees present at University for the help and availability.

Contents

Introduction

Light in the ocean is essential, many forms of life depend on it. Primary producers, such as phytoplankton, use the energy of the light for carbon fixation, visual predators use the light to detect prey, and many organisms, including zooplankton and fish, use light to navigate in the environment. Variation of light intensity, from day to night and seasonal variability in light, is an important regulator of the life of many organisms.

Copepods are one of the most common organisms on Earth (Mauchline 1998). They use light cues to navigate in the environment and light may also regulate their small scale behavior (REFERENCE, this study) Copepods are central and important in the pelagic food web. As new technics and technology develops, our knowledge about their small scale behavior and their interaction with the ecosystem continues to increase(Petzoldt, Lars Rudolf et al. 2009, Maszczyk 2016). My thesis centers on how pelagic copepods, represented by three species of different size respond behaviorally to various controlled light regimes. I hope that my experiment could be useful to improve the knowledge of the marine ecosystem.

Light in the sea

Light is one of the most important components for life on Earth. Light varies between different positions on the planet. This variation is usually due to the different latitude and season, and it is the result of the change in intensity and angle that reaches the surface.

Solar rays are composed of photons that travel in the space with a specific energy, frequency of vibration and length of an electromagnetic wave (λ). The wavelength is measured in a range from 100 to 50 000 nm. Within this range we can classify light as different kind of lights. Visible light is from the 400 to 700 nm, ultraviolet (UV) below 400 nm, Infrared (IR) above 700 nm and far Infrared light over the 5,000 nm. The IR represents the 50% of the light, while UV constitutes 5% and the remaining 45% by visible light.

The sun rays reaching Earth are almost parallel, but after reaching Earth the light loses part of its energy because of the atmosphere and the gas inside it. The surface of the sea also reflects some light, and this reflection depends on the angle of the light that hits the water surface.

The amount of light on the surface of the Earth changes seasonally. The alternating of seasons depends on the rotation of the Earth along its orbit around the Sun. The Earth's axis and its orbit form an angle of 23.24 degree. The spherical shape of the Earth combined with its inclination produces a variation in the intensity of the light on the water surface. When the light reaches the surface with an angle of 90 degrees, the reflection is about 3%. The percentage of reflection increases with the decrease of the elevation's angle of the Sun. When the elevation angle reaches 5 degree, the light reflected is the 45%. This creates an exposure to a different light intensity and the turnover of day and night, creating also a variation in the temperature and climate. When we move closer the poles this variation between different seasons is stronger than at the Equator, where instead it is almost absent because the rays here are perpendicular.

Below the water-surface, the intensity of the light and the composition of it changes, and it is reduced by absorbance, scattering, reflection. The molecules that compose the entire sea all interact differently with the light depending on their composition. 96% of the molecules of the sea is pure water (H_2O), while inorganic and organic matter, such as salts, organic molecules, organisms, constitutes the other 4%. Water readily absorbs IR,it is already absorbed entirely in a few meters of water. Figure 1 shows how absorption of light in the water changes with depth. Light generally decreases exponentially in the water column according to Beer's law $I_z=I_0e^{xz}$, where z is the depth, I_0 is the intensity of the light at surface x is extinction coefficient, and I_z is the light intensity at z depth. The IR and far IR are present at the surface, after the first cm of depth the IR above 1300 nm is absorbed entirely, at 1 m depth the IR is reduced under 900 nm, at 10 m depth remain only the visible light from 300 to 650 nm of wavelength.

All light absorbed is transformed to heat, this produces vapor, maintains the temperature in the water, and creates a flux of water and currents. The light that remains available to the photosynthesis and photochemical reactions is less than 1%.

Figure 1 shows the absorption in an oligotrophic ocean with little organic matter where light can reach a 100 m depth or more. Close to the coast, the interaction with human activity and the river creates a different scenario, where light does not reach as far. The lit zone is called the photic zone.

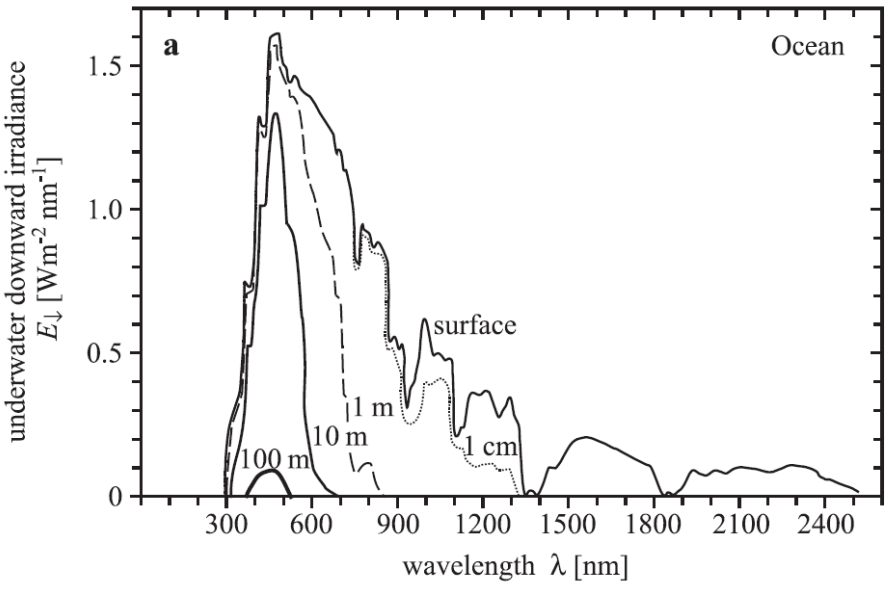


Figure 1: Absorption of light in the ocean. (From Jerlov 1976)

The open ocean and coastal waters are different. The open ocean represents most of our oceans, and is on average 4000 m deep. Surface water are influenced by light and winds, and the surface temperature changes during the year. The deep part of the ocean is stable, and its temperature is cold and does not change. This difference between the two layers creates a deep thermocline that prevents water exchange. The surface of the ocean is often nutrient poor because without mixing of the water the nutrient can not replaced by new, everything from the surface after sink in the deep, remain there. The primary producer needs nutrients from the deeper layers. The only way for the nutrients to return to the surface is by upwelling, that is a current from the bottom of the ocean that arrives at the surface. These upwelling zones have a high production rate because they are very rich in nutrients. The artic is the only place in the ocean where the low temperature in the winter allows the disappearance of the seasonal thermocline and the continuous mixing of the water.

Coastal waters are different compared to the open ocean. Normally the stratification of the water is not stable for the entire year, but in winter with the decrease of temperature in the surface and increased wind forcing, the two layers can mix, such that nutrients from depth can reach the surface. Another source of nutrients is from land via rivers. Degraded organic matter and nutrients from land and human activities enriches the coastal water with nutrients. Sometimes this influence, especially when the human activity is too high, can create an excessive presence of nutrients and generate many problems related to extreme primary production.

The presence of organic matter modifies the penetration of the light in the water in two ways. It alters the depth of light penetration and also the type of light. The dissolved organic material and particles have a specific effect on the absorption of the blue light from the 300 to 450 nm (James and Birge 1938, Sauberer 1945). This creates a variation in the penetration of the type of light in the water. In the open ocean just the blue-green light can penetrate. In coastal water only the green and yellow light can reach the deepest part and the depth decrease to 20 m or less. This value depends on the concentration of dissolved material and particles present in the water. For example, during phytoplankton blooms the high concentration of phytoplankton reduces light penetration.

Light and plankton

Without sun light Earth would be a dark, cold planet without life. During spring when the intensity of light increases, temperature also increases and the water is rich in nutrients from the winter mixing. This combination of factors gives to the phytoplankton favorable conditions to bloom.

The phytoplankton use light energy to convert CO_2 and H_2O into glucose. With the 1% of light that reaches earth, phytoplankton and other algae produce about the half of the oxygen on the planet. The energy fixed by primary producers create much of the basis of the pelagic food web as all heterotrophic organisms depend on this energy either directly or indirectly. Many organisms base their life history strategies on the seasonal variation of temperature and resources. The higher availability of resources and the higher temperature increase fitness and the probability of survival. Phytoplankton is a food source for many organisms, like crustaceans, tunicates, poriferans, cnidarians, molluscs and ciliates, as well as for larvae of

many other invertebrates and vertebrates. Because phytoplankton are largely restricted to the upper layers, it is implicit that most of the food for smaller zooplankton, such as copepods, is also in this layer.

A grazer, such as a copepod, has to move to layers where the phytoplankton is present, while in the same time avoiding to be detected and eaten by a visual predator that also relies on light. That creates a wide range of behavioral responses, migrations, specific life history, and different evolution to avoid predation. Predators in an opposite way have to increase their capacity to detect prey, while not being detected by the prey. All of these adaptations maintain the balance between resource availability, costs and fitness in a positive position.

Copepods and their behavior

Copepods are small crustaceans that live in both the sea and in freshwater. They are the most abundant organism in the world, even outnumbering the insects, and represent 15% of the species known. Copepods can be pelagic, benthic, commensal or parasitic. They range in size from 0.5-20 mm. Their presence in the zooplankton is high, they can represent the 55-95% of the zooplankton, and 25-80% of the plankton biomass (Mauchline 1998). That variability depends on the season, place, and presence of other organisms. Copepods are food for many animals, such as small larval fishes, fishes, invertebrates and whales. Since they are a grazer, they are the direct connection with the primary producers.

Copepods face the same challenges as the rest of us. They need to find food, avoid being eaten by predators, and to find mates in order to survive in the sea. To do all of these thing copepods need to perceive their environments.

Copepods have three different kinds of receptors: photoreceptors, chemical receptors, and mechanoreceptors. The copepods have a simple naupliar eyes composed by two dorsal ocelli and one ventral ocellus (Boxshall 1992). This naupliar eye is not complex enough for image formation, but it is sensitive to slight changes in the light intensity. They use this capacity to regulate their position in the water column (Tranter, Bulleid et al. 1981) and to detect a rapid variation in the light created by some possible predators (Buskey and Hartline 2003).

The chemoreception gives the possibility to smell food, partners, and predators in the area (Heuschele and Selander 2014). Copepods can alter their behavior in relation to such

chemicals signals. These receptors can be pores (Barrientos 1980) or throughout the cuticle (Hallberg, Johansoon et al. 1997). Other form of sensor are bimodal with mechano and chemosensory either, such as in *Temora stylifera* (Paffenofer and Loyd 1999).

With mechanoreseptors copepods can detect the motion of both small and large particles (Kiørboe and Visser 1999). The mechanoreceptors can have different shape (Huys and Boxshall 1991), spines or setae. The setae bend in velocity gradients created by movement of for example a predator (Fields, Shaeffer et al. 2002). Copepods respond to these signals with rapid escape jumps (Titelman and Kiørboe 2003).

Copepods have different type of swimming behavior depending on species, ontogeny, and food availability (Titelman and Kiørboe 2003). For example, the species used in my study behave distinctly different. *Temora* has a constant swimming speed, produced by the movement for catching the food; *Acartia* jumps and sinks all the time without long pause between the jumps; *Calanus* instead between jumps has longer pause during that they sink or swim(Tiselius and Jonsson 1990).

Light stimuli can alter the small scale behavior of copepods (Waggett and Buskey 2007). The kind of reaction depends on the species, the sex, the age of the animal and external signals such as the presence of food or predator. Some use light signals only to control the vertical position in the water column(Cohen and Forward 2002), others show a positive phototactic behavior and go towards the light(Cohen and Forward 2002), while some have a negative response(Cohen and Forward 2002).

As grazers copepods rely on phytoplankton, which in turn rely on light. In terms of risk, light is dangerous. The change between day and night is the basis for the circadian rhythm, which many organisms use to regulate their activity. Many copepods perform Diel Vertical Migration (DVM) where they migrate in the water column between food rich upper layers and deeper darker waters to avoid the predation. Some copepods are nocturnal grazers and go to the surface during the dark to do not be visible to the predator while others have an opposite behavior. For example, *Centropages typicus, Calanopa americana, and Acartia tonsa* (Cohen and Forward 2002, Stearns and Forward 1984b) have a nocturnal migration. Others species like *Anomalocera ornata* (Cohen and Forward 2002) shows a diurnal migration. A third group do not respond to the light and tend to remain in the same depth

Labidocera aestiva (Cohen and Forward 2002). The circadian rhythm is the basis for the migration. The rhythm is persistent without light for days. In some species it can persist for over a month (Svetlichny and Yarkina 1989) while in others species it lasts for only two days (Champalbert 1978, Champalbert 1979).

The vertical migration results from a balance of costs and resource availability. The copepods have a density higher than the seawater, so they tend to sink when not swimming. Svetlichny and Kurbatov (1983) estimated the energy cost in copepods. To remain in a constant depth, copepods spend from 0.02 to 60% of the basal metabolic rate, instead to migrate for 100 m their estimation increase to 13 to 120% of basal metabolism. The quantity of energy required to migrate is high, but the migration allows for avoiding predation. Copepods often regulate the migration with the abundance of food. If the concentration of phytoplankton is high enough at depth, they do not need to migrate to upper layers (Dagg, Frost et al. 1997). In this way copepods may spend less energy for the migration and remain in a darker safer habitat with less risk to meet a predator. In a bright environment the predators, which use their eyes to hunt, life is riskier. The light intensity necessary for the predator to see the copepods depends on prey size, contrast and the irradiance in their habitat – as well as on characteristics of the predator's eyes (e.g. Aksnes and Utne 1997). The *Gobiusculus flavescens*, which live in the coastal water, receives a high intensity of light and the light satiation to see the prey is 5 µmol m^{-2} s^{-1} (Aknes and Utne 1997). For larval cod the value decreases to 1.9 to 7.8 x 10-3 µmol m^{-2} s^{-1} (Ellertsen et al, 1980, Huse 1994).

To increase the survival rate, some species show a plasticity in the coloration related to the circadian rhythm. Vestheim and Kaarvedt (2006) recorded a variation in the coloration for *Parachaeta norvegica* between day and night, where in the dark hours the copepod had a higher coloration compare to the samples taken in the daytime. Other species respond behaviorally by altering feeding between night and day as visible gut contents make copepods more easily detected (Tsuda, Saito et al. 1998)

All in all, the presence of light and its variation influence copepod behavior on both small and large scales. However, only few experiments have quantified these responses in controlled experimental systems where light could be carefully manipulated (Stearns and Forward 1984 b, Cohen and Forward 2002, Cohen and Forward 2005).

Aim and Hypothesis

The overall aim of my project was twofold. First I wanted to develop a setup in which copepod behavior in response to changes in a controlled light environment could be tested. Secondly I aimed to quantify small scale behavioral responses of three copepod species (*Acartia tonsa, Calanus finmarchicus and Temora longicornis*) to different intensities of light. Because prey visibility is intimately connected to prey size and contrast I tested copepod species differing in these respects.

I aim the possibility to develop and build a setup where to test different species of copepods with the possibility to use several light sets. The advantage of automates all process without the possible influence of human presence. In the end gives the faculty to be movable and transportable in other place.

For the second aim I hypothesize a change in the position of the copepods is related to different light intensities. The variation in swimming speed is the same for all species in relation the variation of the light. Another hypothesis is the bigger of three species shows a stronger reaction to the light stimuli.

Material and methods

The methods section is divided into two parts. In the first part, I describe the development of the experimental protocol for measuring behavior in response to irradiance. In the second part, I describe the experiment that was designed to test the effect of irradiance on copepod behavior.

Filming setup development

To study the behavior of the copepods in response to irradiance, I designed and built a filming setup dedicated for the purpose. I used the following criteria to design and build the final setup:

1. The system should be easy to mount and clean.
2. Exchange of water and animals should be easy and require little time
3. The copepods should be able to move freely inside the tank without any obstacles
4. Only the experimental light should be present inside the tank
5. The quality of video should be sufficient for an automatic tracking system
6. Experimental light should be controllable and well defined

The process to develop the system, hardware and software, required many steps with different grade modifications. Below I describe this process. I attached an SOP (Standard Operating Procedure) in Appendix for the setup.

Hardware development

I want to give the copepods to have enough space to move and migrate in response to the light. The first idea was a classic rectangular tank in glass, but the corners created reflections and attracted animals (pers. obs). The best solution, therefore, appeared to be a cylindrical tank. The tank was made by the UiO/MN Instrument Workshop (http://www.mn.uio.no/tjenester/instrumentverksted/); it is a cast acrylic tube of 50 cm length and about 10 cm in diameter. A cast acrylic tube is stronger and more transparent than an extruded one, but can vary slightly in diameter. To ensure a bubble-free environment, the tank was closed with two custom made lids with o-rings. After a few trials, one of these lids was glued to avoid potential leakage.

The refractive index of acrylic is closer to water than to glass. The refractive index of water is 1.33, acrylic is 1.49, and glass is 1.62 (Wikipedia 2017). The roundness of the tube creates reflection and distortions, which hampers filming and detection of the copepods. This was solved by placing the observational tank in an outer glass tank (60x12x11 cm) filled with the same water as inside the tank. This removed the distortion, reduced convection in the tank and improved the final images. To avoid rolling of the experimental tank inside the glass tank, I designed a holder for the experimental tank. The holder was printed with a 3D printer (Ultimaker 2+). It went through different steps of evolution (Figure 2).

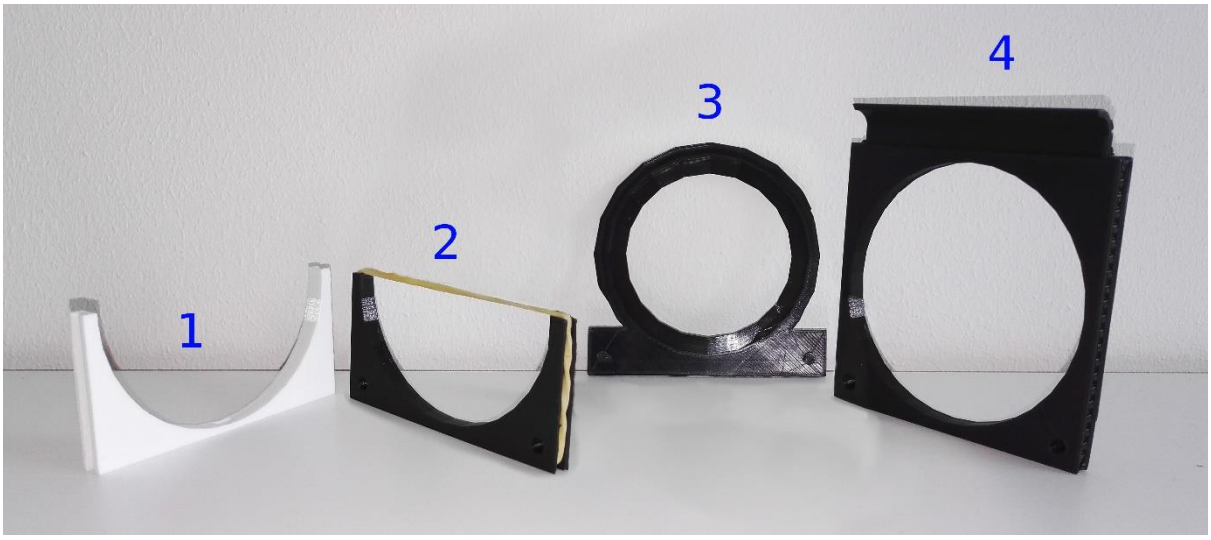


Figure 2: Development of the tank holder from early prototypes (1) on the left to the final version on the right (4). The first version (1) was white but reflected too much light, the second version maintained the same shape but was black (2). The final version consisted of a closed circle with a handle on the top (4) and a round cover to reduce reflections (3).

To ensure that the system was not exposed to unwanted light, I inserted the aquaria in a wooden box painted with blackboard paint on the inside, and covered by a double layer of black plastic on the outside. I covered all LED lights from the electronic parts. The light supply for the recording was provided by an infrared (IR) LED strip (860 nm wavelength). This wavelength is not visible to the copepods and does not influence their behavior ((Stearns and Forward 1984 a),(Cohen and Forward 2002)).

The light for experiment needs to be easily controllable and well defined such that the spectrum does not change with light intensity. The light was provided by WS2812 LED packages, also called NeoPixels (Figure 3). The red, green, and blue LEDs of this RGB device can be controlled individually in 256 intensity steps. I used a microcontroller (i.e., Arduino nano, www.arduino.cc) and a voltage of 5 volts to control the NeoPixels. The spectrum in this LED remains the same across the whole range because the intensity is controlled by Pulse Width Modulation (PWM).

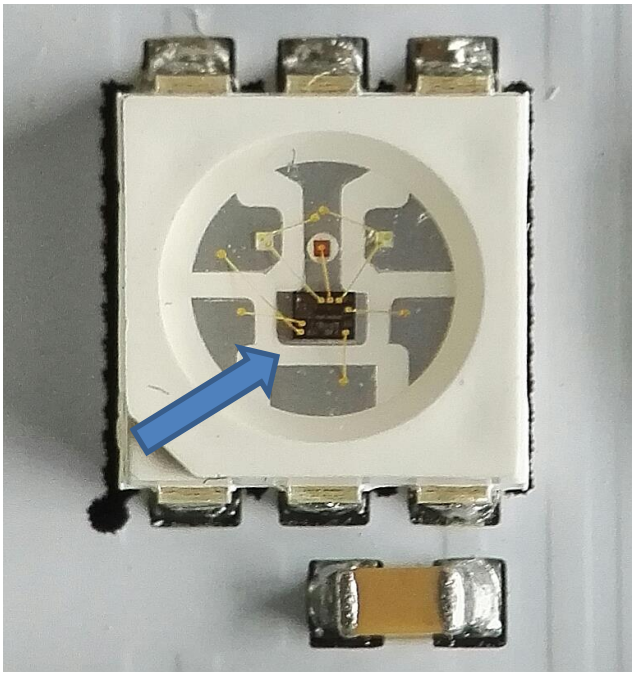


Figure 3: WS2812 or NeoPixel. RGB LED with a small chip inside (arrow) that allows for controlling many LEDs and the intensity of the three colors with a single wire.

I measured the light intensity with a SpectraPen (Photon Systems Instruments). I took two measurements, one behind the lid and the other from the bottom of the tank filled with the water (Table 1). I used this formula to calculate the extinction coefficient (k, cm^{-1}):

$$k = \frac{ln(I_0) - ln(I_D)}{depth(cm)} \qquad (1)$$

where I_0 is the intensity of the light from the led behind the lid ($\mu E/m^2/s$); I_D is the intensity of the light at the end of the tank ($\mu E/m^2/s$); *depth* is the length of the tank (50 cm) . I could

then calculate the light intensity at a given distance from the light source with the following formula:

$$I_c = I_0 e^{-kc} \quad (2)$$

where I_c is the intensity of the light at the specific distance to the light (µE/m^2/s); k is the extinction coefficient (cm^{-1}); c is the distance to the light source (cm).

Figura 4: Experimental tank. Blue depicts the lid that close the tank. L is the light source. Light was measured at position 1 and 2.

Table 1: The first three columns contain the red, green, and blue color value sent to the arduiono, I_0 is the intensity behind the first tank lid, I_p is the intensity value from the bottom of the tank filled with the water, k is the Extinction coefficient. (cm^{-1})

NeoPixel RGB values			I_0 (µE/m^2/s)	I_p (µE/m^2/s)	K (cm^{-1})
R	G	B			
20	20	20	2.67	0.86	0.02
15	15	15	1.80	0.57	0.02
10	10	10	0.95	0.28	0.02
5	5	5	0.39	0.09	0.03
1	1	1	0.06	0.00057	0.09

The video was recorded by two Pi Cameras with 5 Megapixel (mpx) of resolution, without IR filter and in full HD quality. Each Pi Camera was connected to a Raspberry Pi 3 single-board computer running a version of Linux Debian operating system for the ARM processor. (www.raspberrypi.org). The video was saved on external hard drive to avoid frame shifts caused by delays during the recording. After each experiment I photographed the copepods on a special glass slide with a frame made by 3D-printer (Figure 5), using a Canon EOS 7D with a Canon MP-E 65 mm macro lens and a ring flash.

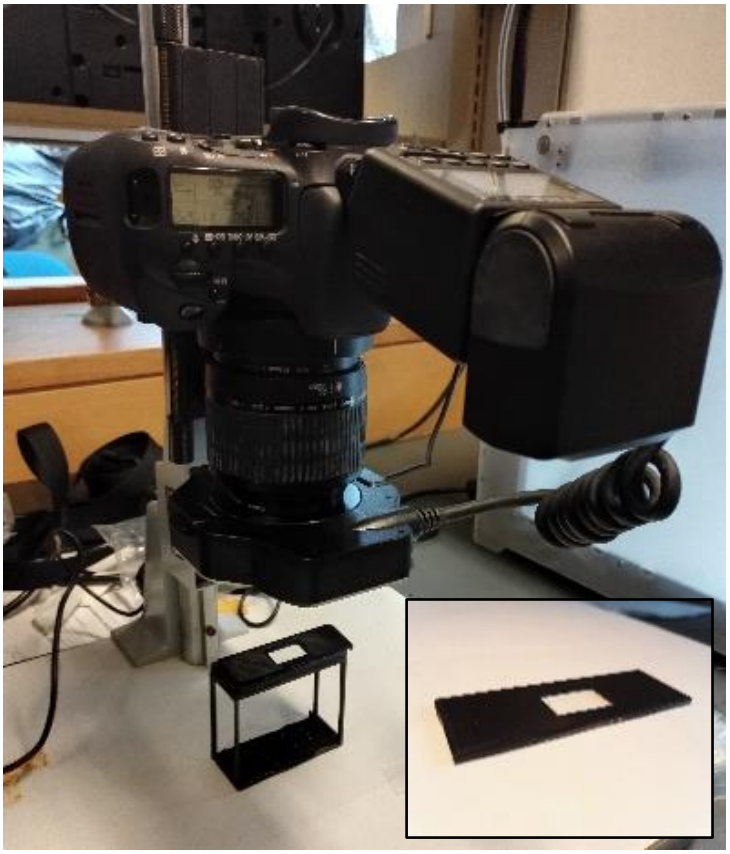


Figure 5: Photographic setup, with the glass slide holder and the 3D printer mask glued on the glass slide.

Software development

The code development included several different environments: Arduino, Raspberry Pi 3, tracking program and data analysis. Arduino has a specific programming language and programming environment to compile and upload the code to the microcontroller. The small code on the Arduino board received the time and light intensity for the experiment by a USB cable connected to Raspberry Pi 3. Interacting with the NeoPixels requires a specific Arduino library created by Adafruit Industries (www.adafruit.com).

The code to control the recording and the tracking were written in Python (Python Software Foundation, www.python.org). Python scripts on the Raspberry Pi 3 control the time to recording in dark and light. It reads the setup from a file on the board, sends to Arduino the setup of the light for each round. The video recorded from the Pi camera was saved on the external HD connected to the board. The second board recorded the video from the other Pi camera. Each file received a name where was specified date and time, group, round, and board. In figure 8 there is a schema of the species, groups, and rounds of the experiment. The two boards controlling each of the Pi Cameras were connected with an Ethernet cable, and one was set up as DHCP server to create a small LAN (Local area network) between the two boards. It is possible to find all code and script in the appendix.

Experimental organisms

The copepods were collected from Oslofjorden (Figure 6), using the research vessel Trygve Braarud. We used a WP2 net (figure 7), with a mesh size of 200 µm and a diameter of the aperture was 57 cm. The net was pulled up from the 195 m depth to the surface. On board of the ship the copepods were immediately transferred to a cooler, which was aerated with an air pump to ensure sufficient oxygen supply during transport.

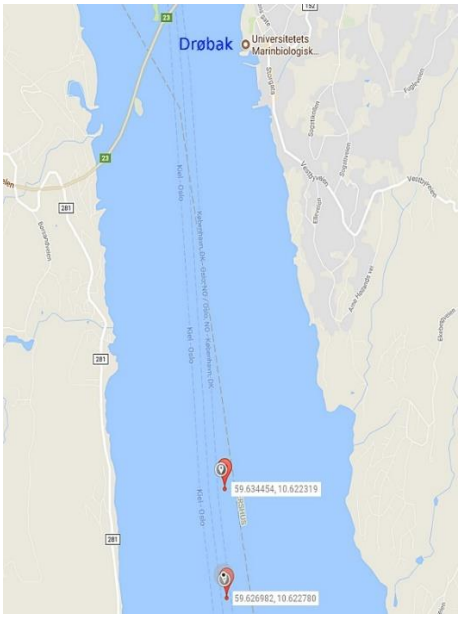


Figure 6: The positions where the copepods were collected, south of Drøbak

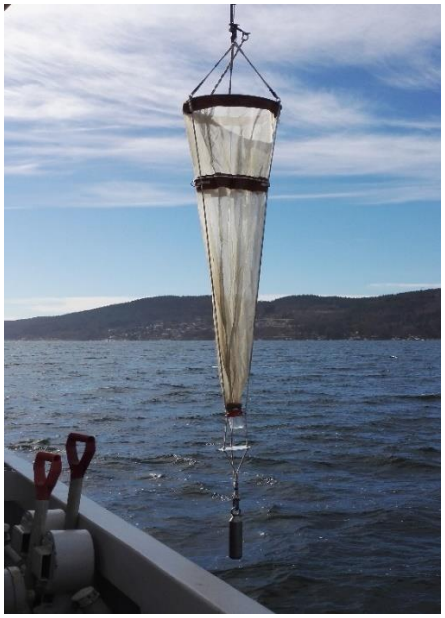


Figure 7: WP2 net ready to collect the copepods

Back at the laboratory, copepods of different species, specifically *Acartia tonsa, Temora longicornis and Calanus finmarchicus* (figure 8), were separated live under a dissection microscope. Copepods were put in three beakers, one for each species, with GF-F filtered salt water (GF-F Whatman filter size pore 0.7 µm, 30 PSU). The copepods were kept in a cold room at 12°C with a 12:12 hour dark:light cycle . Copepods were fed *ad libitum* every two days with a mix of three different algae: *Dunaliella sp.*, *Isochrysis galbana* (5-6 µm, naked motile), and *Rhodomonas salina*.

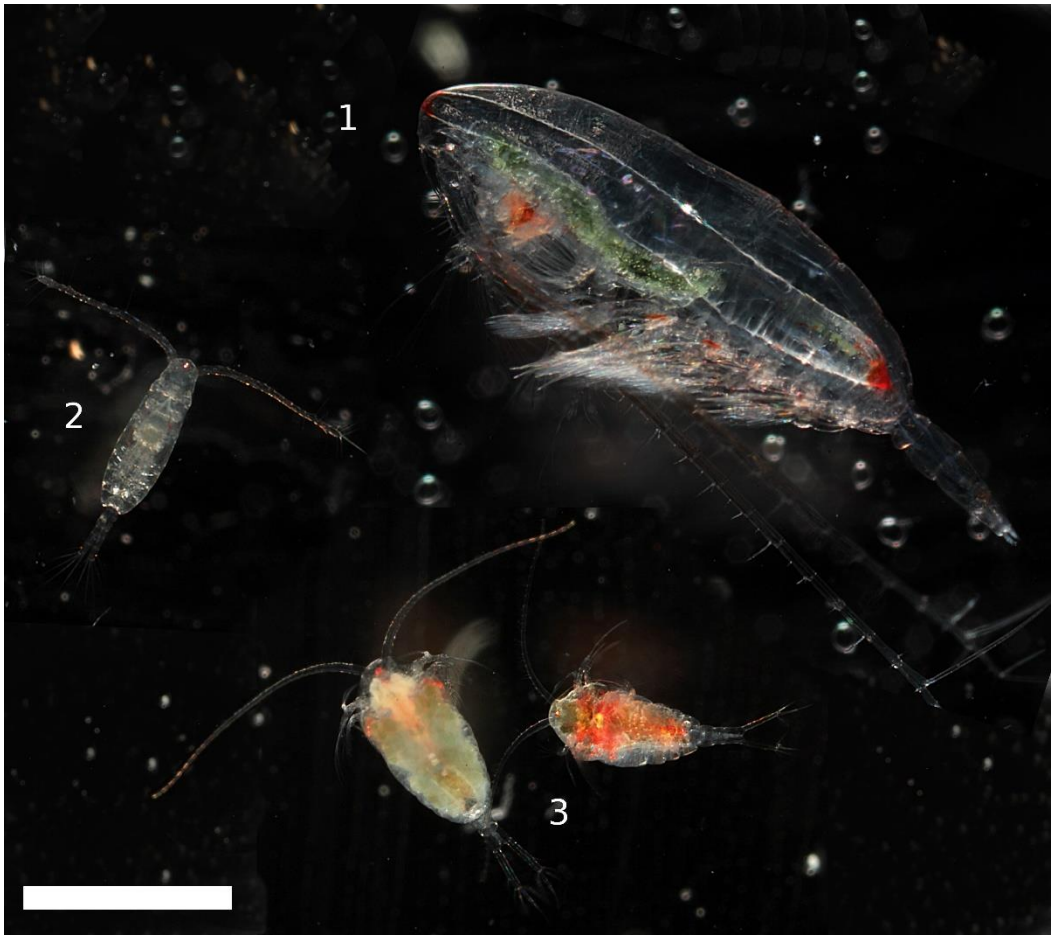


Figure 8: The three species of copepods used for the experiment. All three belong to the order Calanoida. 1) *Calanus finmarchicus* Size: 2.4 – 5 mm 2) *Acartia tonsa* Size: 1.0-1.5 mm 3) *Temora longicornis* Size: 1.2-1.5 mm. (Nielsen and Hansen 1999) Scale bar 1mm. Photo: Pietro Antonio Grassi

Experimental procedure

I designed a series of trials to test the effect of irradiance on the three species of copepods using the setup described above. The general structure of the experiments is illustrated in figure 9. For each of the three species I used 25 groups of 5 animals per group. Each group received five different light intensities (table 1) in random order generated by a Python script.

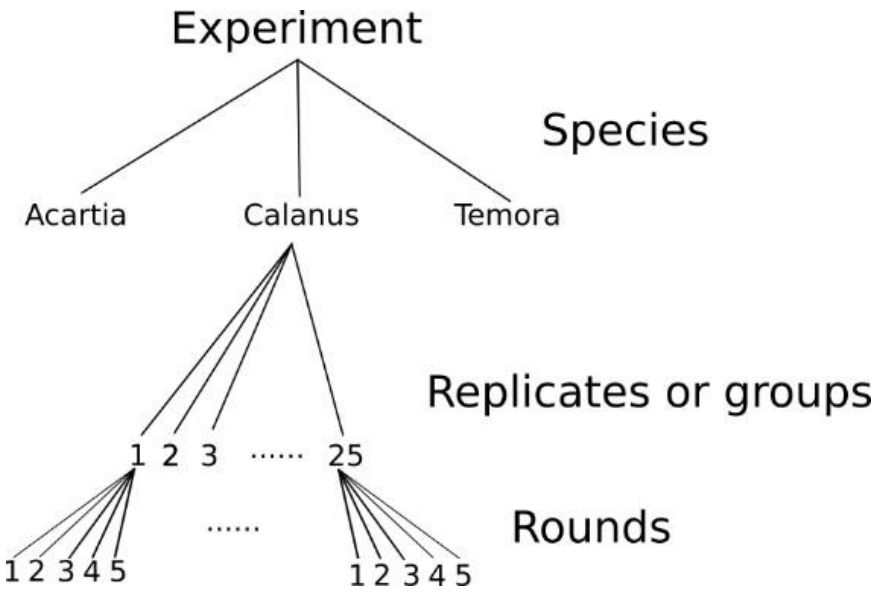


Figure 9:

Hierarchical structure of the experiments

I took a group of 30 animals randomly from the experimental culture beaker and distributed them 5 copepods for each well into a 6-well plate. I placed the 6-well plate in the same room of the experiments to equilibrate to the same temperature. With a 2 ml pipette modified with a bigger hole, to do not damage the copepods, I inserted the animals in the tank previously filled with the filtered water and cleaned from air bubbles on the tank wall. I closed the lid gently and with care to avoid capturing air inside the tank. I then brought the tank in the experimental room and placed it inside the glass tank. The IR light was placed over the glass tank, and I closed the box with the plastic cover. After the box was ready, I inserted the light value for the specific experiment to the file on the main Raspberry Pi 3, saved it and ran the python script for the experiment. The box was put in a climate room without any windows. The temperature in the tank was between 12-14 ºC during all experiments. To enhance work

flow, I used two experimental tanks and holders for them. This gave me the possibility to change water and animals while simultaneously filming another trial.

Figure 10 shows one example of the experiment for a single group of copepods. The first step is 15 minutes of adaptation to the tank in the dark without recording. I inserted two seconds of increasing light to avoid scaring the animals with too fast light change. I used 25 groups of animals with a randomized order of light intensity. I wrote a small python script to create a random table with the value for all 25 groups for each species. Between each light and dark period, I inserted 3 seconds of pause to start a new video. For every group, I obtained 5 videos.

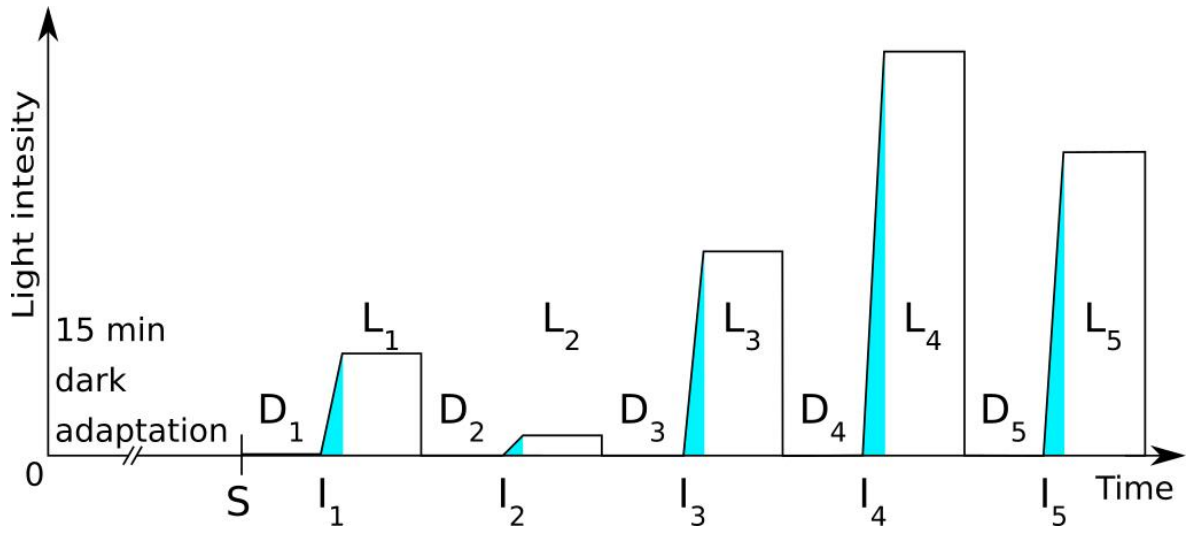


Figure 10: Schematic figure of periods of light exposure in the experiment. 15 minutes in the dark, without recording. S – start recording. D_n is dark. I_n is 2 seconds increase light from 0 to the value. L_n is light, subscripts 1-5 denotes order in round experiment. In the example, the order for the light values are 5, 1, 10, 20, 15. The number indicates the exposure round.

I took pictures of all animals after each experiment, and measured their prosome lengths with ImageJ (imagej.net/fiji) from these pictures. The dry weight (DW) is the estimated from the prosome length using the formula:

$$DW = 10^{(B \log Length) - A} \qquad (3)$$

Where *Length* is the length of cephalothorax, *A* is regression coefficient, *B* is the exponent (Hay, Kiørboe et al. 1991). From individual DW estimates, I calculated the average dry weight of each group.

Movement analysis

A python program is based on OpenCV library (Open Source Computer Vision Library ver. 3.2) was used to analyze animal movements in each video sequence (figure 11). The program converts each frame in black and white and compares in the beginning a certain number of frames, fixed by me at 150, to create the background to subtract to the other frames. The program has a main window with the possible variable, the dimension of the object, maximum jump and different regulation of the shape. After the selection of the video, the program creates three new windows: the first with the original video, where it marks found copepods with a red circle; the second with the converted video in black and white and the third with white background after it had subtracted the background from the video. Positions of each animal during the experiment are stored in a text file.

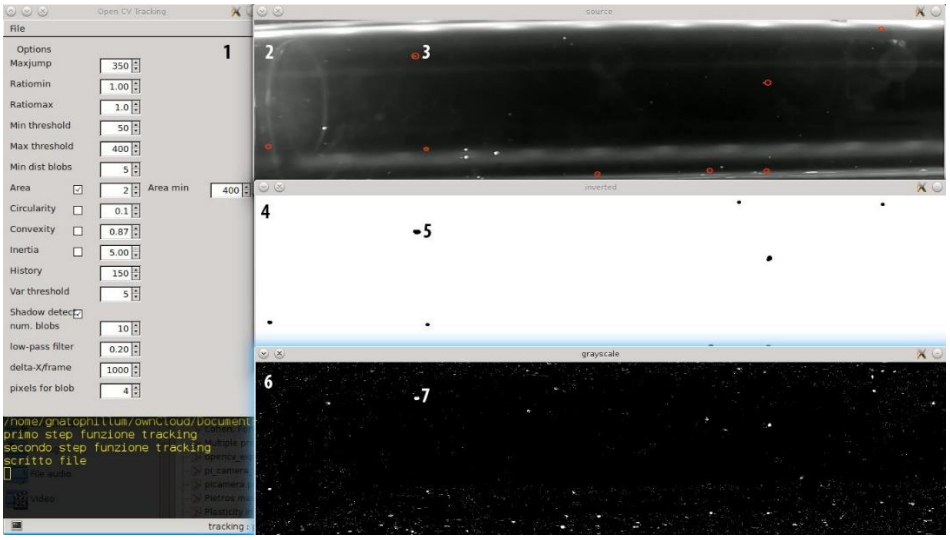


Figure 11: Screenshot of tracking program.
1 graphic interface
2 Original video
3 Tracking
4 Negavite screen
5 Tracking, same of 3
6 Background to subtrack
7 Tracking, same of 3 and 5

Statistical analysis

The statistical analysis was made with R (ver. 3.3.2). The collected data was rearranged and I calculated the distance in cm, the intensity of the irradiance in all length of the tank, and created a filter to remove all possible errors in the tracking. I used the libraries dplyr (Ver. 0.7.3), nlme (3.1-131), visreg (Ver. 2.4-1), ggplot2 (Ver. 2.2.1).

The general graph is a view of the tracking for the single group divided by the rounds (Figure 9) Each combination of species, replicates and round was given a unique ID.

It was tested the influence of the time on the distance with the species and round as random factors.

I calculated the average distance from the light source in cm, and its corresponding standard deviation and standard error, for each combination of species, group, and round.

I used the formula 2 to calculate the light intensity perceived at specific distance to the light source.

It was tested the relation between the dependent variable distance to the light with the independent value species with a random factor as the round.

To test the influence of light intensity on the perceived light intensity for each species I used a linear mixed effects model, where the perceived light intensity is the dependent variable and the light intensity is the independent, and species and round are random factors.

I used the logarithmic to transform and give more linearity to the mean speed, use it as dependent variable in relation with the independent variable the intensity of light and as random factor the species.

A similar test was calculated to control if there is a significant variation between the dark and light exposure. The dependent value was the logarithm of mean speed in relation with the independent value species, and the light condition and round as random factors.

We always started with a full model including all factors and interactions and selected the best model by using stepwise backward deletion. To account for repeated measurements, we

included the unique group id as a random factor in the model. We confirmed model fit by examining the distribution of residuals of the final models.

If a factor or an interaction had a significant p-value in ANOVA table (below 0.05), the included single factors were retained in the model (Sokal and Rohlf 1995).

Results

Body weight

The three species show a different size, colour, and transparency. The distribution of body weight for each species as calculated from the prosome length is presented in figure 12. Calanus has an average weight of 225.73 µg. They are the biggest species in the experiment, and show a big variation in the size. Acartia and Temora have similar length, but Temora has higher dry weight compared with Acartia, because the body is wider than Acartia. Temora has an average weight of 21.38 µg and the Acartia 5.08 µg; both species have small variation in body sizes (Table 2).

Table 2: Mean Dry weight values for the three species with the SD, and coloration.

Species	Mean Dry Weight (µg)	Standard deviation	Coloration
Acartia tonsa	5.076	1.01	Low
Calanus finmarchicus	225.73	115.79	High
Temora longicornis	21.38	5.06	Low-Medium

General behavioural patterns

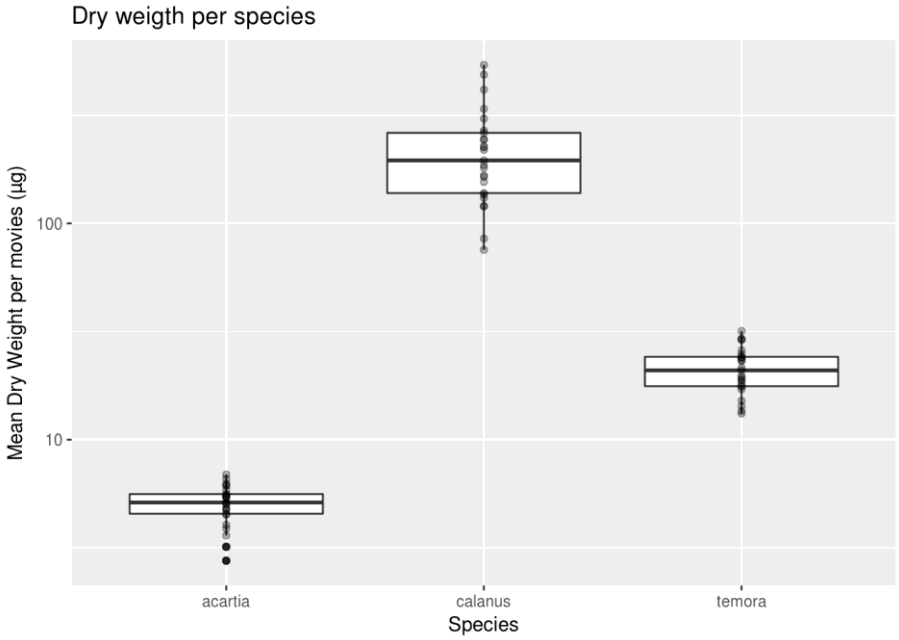


Figure 12: Distribution of the weight of the three species used.Y-axis is on logarithmic scale.

The tracking program produces a series of positions for each round. In figure 13 there are three examples of representative tracks, from which it is possible to see how the animals change their position after the illumination started (the black vertical line represents the onset of illumination). In the first graph the distribution of *Acartia* shows that the distance of animals to the light source decreases after every round, with the exception for few single points that are probably false tracks.

For *Calanus* the pattern is different from the *Acartia* (figure 13b). In the graph, it is possible to see two different behavioural patterns. First the distribution before and during the exposure is different, in the dark exposure the copepods are deployed in all tank; after the line when the illumination begins, the copepods are almost all of them on the other side of the light source except in the group 6 (the green dots) where some animals go in the opposite direction, Second, the copepods after every round are more concentrate in the half tank at the opposite side of the light source. The last figure 13**c** shows *Temora* in the first round, they are distributed in all tank, but after the starting of the illumination, most of them go directly to the other side of the light source. In the next round, the presence of animals in the center of the tank is almost absent. However, there are some animals close to the light source.

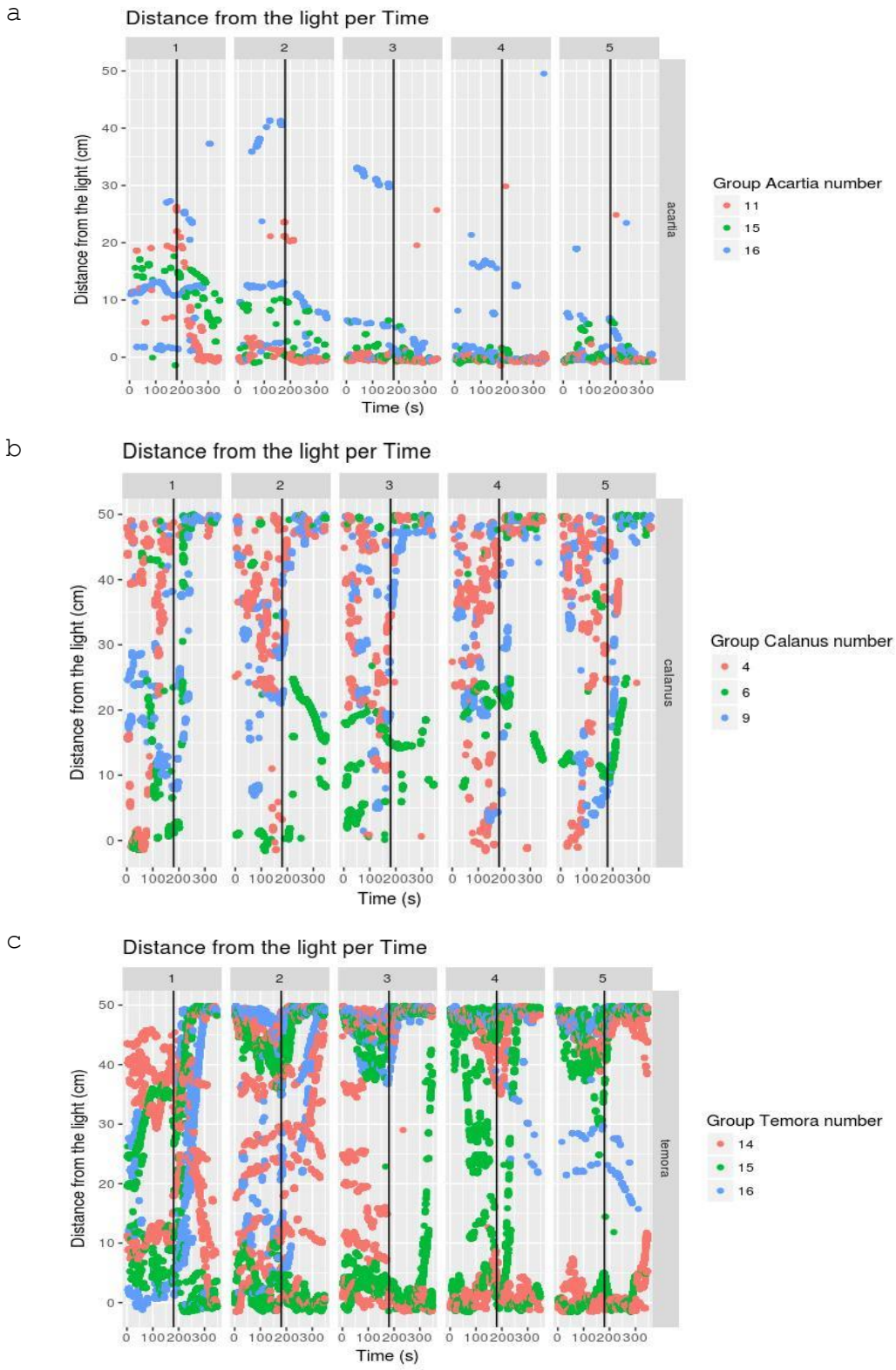


Figure 13: Representative tracks from three replicates of each species. The black vertical line represents the passage from dark to lightness.

Distance to the light source

The copepods are sequentially exposed to five different irradiation level during the experiment for five rounds. The analysis of the data shows that the distance from the light source for each species is related to exposure round (Table 3). In figure 14 we can see how the *Acartia* changes their average position every round, they move closer the light source. The difference between each position and the next one became smaller with the increasing of the round. *Acartia* shows a photophilic behaviour. In contrast *Temora* and *Calanus* change their position in the opposite direction of the light source, showing a photophobic behaviour. While *Calanus* increases the distance to the light consistently, *Temora* shows a big jump after the first exposure round, followed a more stable movement away from the light source.

Table 3: ANOVA and summary table for the dependence of distance to species and exposure round

	numDF	denDF	F-value	p-value
(Intercept)	1	670	35.981	<.0001
Species	2	72	12.108	<.0001
Round	1	670	22.128	<.0001
Species:Round	2	670	31.130	<.0001

(Intercept)	12.695	2.116	670	5.998	0.0000
Speciescalanus	9.928	2.997	72	3.311	0.0015
Speciestemora	14.389	2.993	72	4.807	0.0000
Round	-1.658	0.352	670	-4.704	0.0000
Speciescalanus:Round	3.569	0.501	670	7.128	0.0000
Speciestemora:Round	3.232	0.499	670	6.483	0.0000

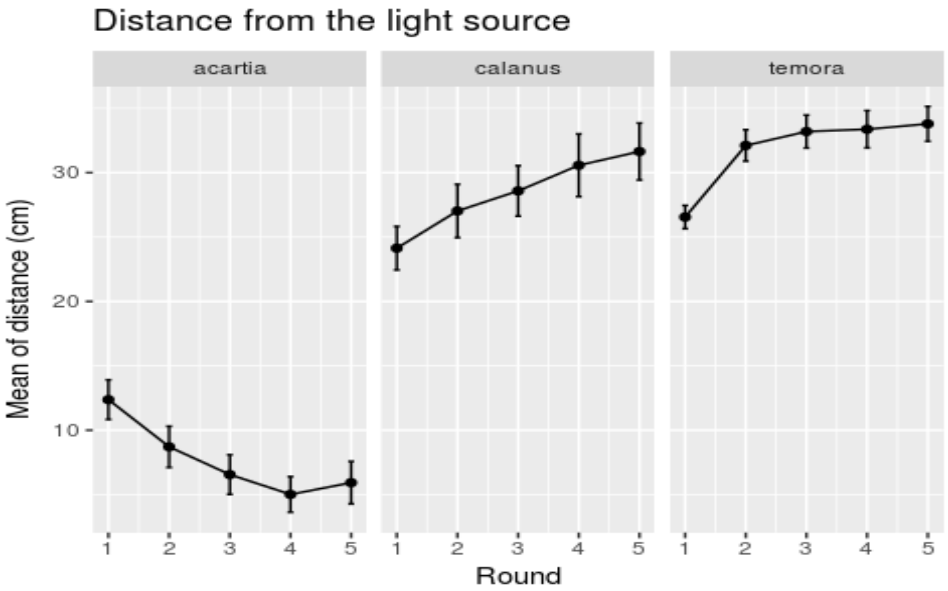


Figure 14: Position (cm) ± SE of each species for the exposure rounds

Perceived light intensity

The perceived light intensity for each species is influenced by an interaction of exposure round, species and light intensity (Table 4). Figure 15 shows the relationship between the perceived light intensity and the intensity of the irradiation for each species during each round. *Acartia* maintains the behaviour previously showed, they go towards the light source. Their perceived light intensity is close to the maximum possible during the exposure. This is visible when comparing the value for the species within the light-blue bar that represents the range of light behind the lid (maximum light) and the bottom of the tank (minimum light). In contrast to *Acartia*, *Temora* and *Calanus* tend to remain in the lower half of the light range after the first round.

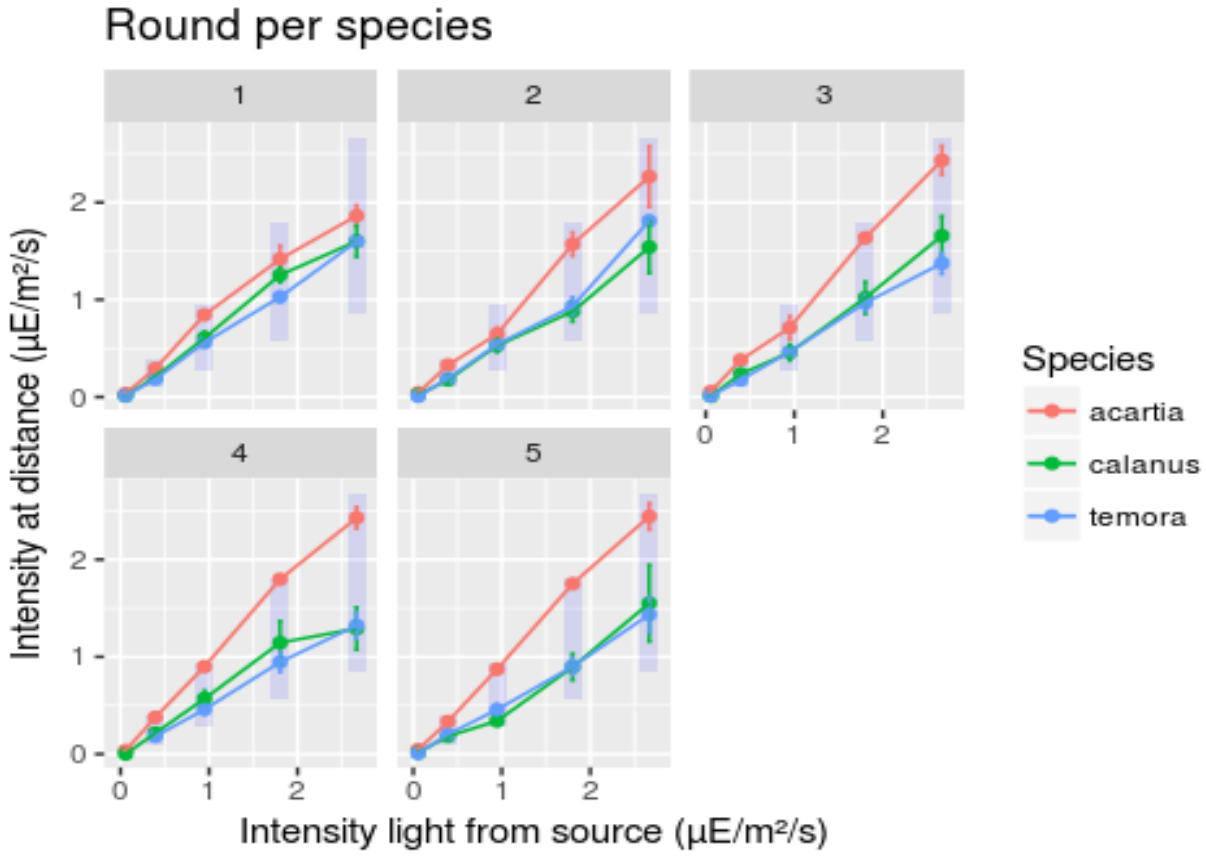


Figure 15: The average perceived light intensities ± SE as a function of the intensity of the light source, for each of the five exposure rounds and for each of the species. The blue boxes inside the graph represent the range of possible irradiances for the exposure levels in the experiment.

Table 4: ANOVA and summary table for the dependence of perceived light intensity on light intensity, species and round.

	numDF	denDF	F-value	p-value
(Intercept)	1	291	0.003	0.9545
Intensity	1	291	273.365	<.0001
Species	2	72	0.151	0.8600
Round	1	291	0.005	0.9386
Intensity:Species	2	291	2.608	0.0754
Intensity:Round	1	291	9.147	0.0027
Species:Round	2	291	0.200	0.8182
Intensity:Species:Round	2	291	6.403	0.0019

	Value	Std.Error	DF	t-value	p-value
(Intercept)	-0.0040	0.0692	291	-0.057	0.9545
Intensity	0.7457	0.0451	291	16.534	0.0000
Speciescalanus	0.0166	0.0981	72	0.167	0.8664
Speciestemora	-0.0357	0.0971	72	-0.367	0.7144
Round	0.0015	0.0191	291	0.077	0.9386
Intensity:Speciescalanus	-0.1267	0.0637	291	-1.987	0.0479
Intensity:Speciestemora	-0.1265	0.0645	291	-1.962	0.0507
Intensity:Round	0.0413	0.0136	291	3.025	0.0027
Speciescalanus:Round	-0.0132	0.0276	291	-0.478	0.6333
Speciestemora:Round	0.0037	0.0275	291	0.136	0.8923
Intensity:Speciescalanus:Round	-0.0543	0.0194	291	-2.806	0.0053
Intensity:Speciestemora:Round	-0.0650	0.0197	291	-3.310	0.0011

Swimming speed

The swimming speed of the copepods during the experiment was influenced by an interaction of light intensity and species (Table 5). In this analysis we only included data acquired during light exposure, without the dark exposure. In figure 16 it is possible to see the three different patterns of swimming speed. *Acartia* shows the highest mean speed with the lowest light intensity value, but they reduced it quickly with increased light levels. At high light intensities the mean speed is almost stable. *Calanus* shows a variation between the first and second light intensity, but after the third the mean speed was maintained lower and more stable than in the first and second rounds. *Temora* instead shows a very low variation in their speed during all experiment.

Table 5: ANOVA and summary table of the dependence of swimming speed on species and light intensity

	numDF	denDF	F-value	p-value
(Intercept)	1	295	151.230	<.0001
Intensity	1	295	23.793	<.0001
Species	2	72	73.690	<.0001
Intensity:Species	2	295	5.491	0.0046

	Value	Std.Error	DF	t-value	p-value
(Intercept)	-1.114	0.091	295	-12.300	0.0000
Intensity	-0.292	0.060	295	-4.878	0.0000
Speciescalanus	-0.965	0.128	72	-7.552	0.0000
Speciestemora	-1.533	0.128	72	-12.013	0.0000
Intensity:Speciescalanus	0.220	0.085	295	2.604	0.0097
Intensity:Speciestemora	0.260	0.085	295	3.080	0.0023

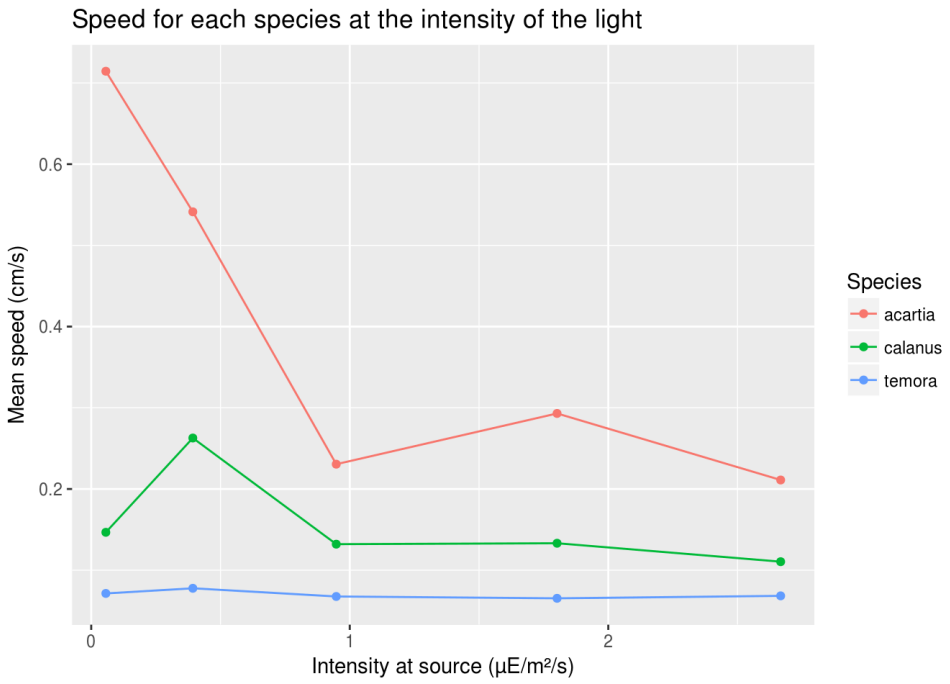


Figure 16: The difference of the mean speed for each species

When we compare swimming speeds between light and dark conditions, we see is a not significant interaction between the speed and the different light condition for each species and round (Table 6). To arrive at this result I started with the relation between mean speed and species with round and light condition removing the not significant factors with the backword

Table 6: ANOVA and Summary table of the final model describing the depende of mean speed on species, round and the light condition

```
                numDF denDF  F-value p-value
 (Intercept)        1   671 378.018  <.0001
 Species            2    72 192.808  <.0001
 Round              1   671  13.874   2e-04
 LightCondition     1   671  16.624   1e-04

Fixed effects: log(MeanSpeed) ~ Species * Round *
LightCondition - Species:Round:LightCondition -
Round:LightCondition - Species:Round - Species:LightCondition

                        Value  Std.Error  DF    t-value p-value
 (Intercept)          -1.445    0.075    671    -19.443   0e+00
 Speciescalanus       -0.677    0.064     72    -10.495   0e+00
 Speciestemora        -1.262    0.064     72    -19.620   0e+00
 Round                 0.067    0.018    671      3.725   2e-04
 LightConditionLight -0.205    0.050    671     -4.077   1e-04
```

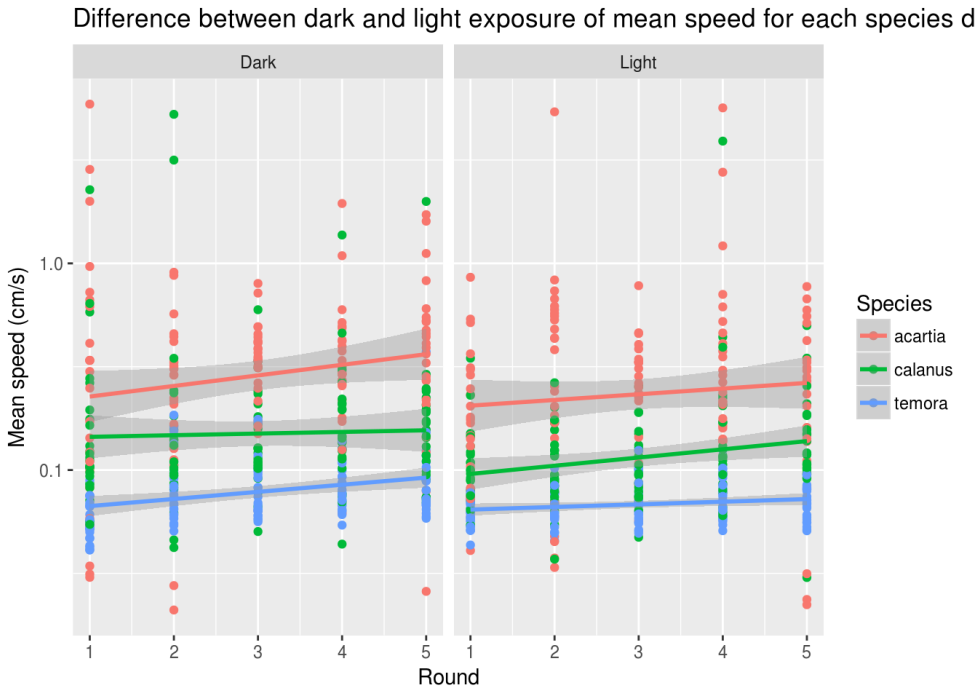


Figure 17:

The mean speed (cm s^{-1}) for the dark exposure and light exposure

Discussion

Design the setup

In my master thesis project, I designed and constructed an experimental setup capable of measure the behavioral reaction of copepods to changes in light conditions. Similar studies had been done before.

Table 7 summarizes published studies on the influence of light on copepods, the species used and the dimension of the tank. The studies with *A. tonsa* used a tank with minimum length about 10 cm. The illumination used in these studies were different. For my setup, I used a LED NeoPixel, the simplicity to regulate the intensity and the quality of the light made them an optimal source of light. It is possible to control all system light, recording, by one only electronic system, without a mechanism or motorized part. The combination of Raspberry and Arduino gives a simple way to automatized all process with a minimum human interaction and without vibrations or sound produced by a mechanical system. Another advantage of my system are the low energy requirements and the small dimensions of the setup. Raspberry, Arduino and NeoPixel work with a low voltage and a low quantity of current, which makes this setup portable and potentially powered by a battery.

Common to many of the previous experiment done on copeopds and light are the smaller dimensions of the tank. Copepods can however have a swimming speed about 1 – 5 time the body lengths s^{-1} (Mauchline 1998). Buskey (1994) measured the swimming speed of *A. tonsa* in 1-10 mm/s; Hardy and Bainbridge (1954) obtained for upwards 18.3 mm/s in downwards 29.7 mm/s for *C. finmarchicus*, for the *T. longicornis* the upwards speed was calculated in 5.3 mm/s.

The reaction to the light exposure has a duration about 10s (Stearns and Forward 1984 b), after this period the copepods turn back to normal behavior if the intensity of the light does not change again. In my setup I try to give the possibility to the copepods to overpass this 10 s swimming in one direction, so it is possible to control that the animals after this time still show the behavior related to the stimuli or back to the previous one.

Table 7: The table summarize some studies on the influence of light on copepods

Ref.	Tank dimension	Light	Time	Species
(Stearns and Forward 1984 a)	15x3.5x3.7 cm	300w or 750w	1min dark 75 s light	*Acartia tonsa*
(Stearns and Forward 1984 b)	3x1x25 cm		21 stimuli and 30s between every decrease of light	*Acartia tonsa*
(Cohen and Forward 2002)	3x3x5, 5x5x5 cm	400w HQL	5 s exposure, 3 min dark	*Centropages typicus, Canaopia americana, Anomolacera ornata, Labidocera aestiva*
(Cohen and Forward 2005)	4x4x5 cm	750w I.l.	20s light 6s after stimuli	*Calanopia americana*
(Aarseth and A. 1999)	100x9Ø cm	UV and white light tube 30w	6h dark, 1h light Vis. and Vis.+UV	*Lepeophtheirus salmonis, Calanus finmarchicus*
(Stearns 1986)	10x10x10 cm	300w tungsten filament halogen lamp		*Acartia tonsa*
My experiment	50x10Ø cm	NeoPixel led	3 min dark, 3 min light	*Acartia tonsa, Calanus finmarchicus, Temora longicornis*

Position in response to light exposure

All three species show a reaction to the light in the experiment. As shown in the result (figure 14) *A. tonsa* is positively photophilic and *T. longicornis* and *C. finmarchicus* are negatively photophilic. The variation of the mean position for all group of each species change from closer to the center to one of the sides of the tank, dependent of the exposure rounds. This means that the copepods moved toward or away from the light source while being exposed. But when the exposure finished they displayed a different behavior. During the dark phase, *Acartia* remains almost in the same zone, *Calanus* goes back to almost the previous position after each stimulus, *Temora* after the exposure goes back but remains in the middle of the tank (Figure 13). It is easy to think that the average position for each group for every intensity

of light sometimes does not capture the differences for each exposure. Figure 15 shows the difference between the perceived light intensity at each position, expressed as a function of each light intensity of the source.

The copepods during the first exposure round take a position around the medium value of the possible light intensities in the aquarium. In the next rounds, the position shifted progressively to the sides of the tank. After the third round, the position of *A.tonsa* remain close to the light source, *T. longicornis* and *C. finmarchicus* remain in the two third of the tank from the light source. This is in accordance to Stearns and Forward (1984 a), who found a positive phototactic from the *A. tonsa* with a threshold of $2.8x10^{11}$ photons $m^{-2}s^{-1}$($6.02x10^{17}$photons=1 microEinstein= 1 micromol). Lower value from the light source in the experiment is 3.61 $x10^{16}$ photons $m^{-2}s^{-1}$and from the far side is 2.06 $x10^{13}$ photons $m^{-2}s^{-1}$, so the minimum value in our experiment is higher than the threshold value evoking a response in this species. In another study of the same authors (Stearns and Forward 1984 b) the copepods showed a correlation between the upwards response towards the light and the decrease in light intensity (figure 18), thus *A. tonsa* shows a correlation between the light intensity and the position in the water column. But in my experiment, the different light intensities did not correlate with the position. This might be explained by differences in the experimental procedures in our experiments. Stearns and Forward gave a continuous light and modified the intensity, while in my experiment the different exposures where continues but were interrupted by a dark period. They notice that the copepods maintain the upward movement only for a short period. This suggests that phototactic behavior is not correlated directly to the intensity, but a short-term response to a modulation of the light intensity.

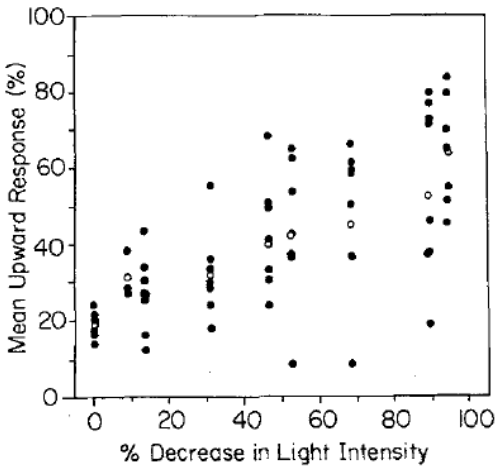


Figure 18: Example of vertical behavioral response to light for *Acartia* (Stearns and Forward 1984 b)

C. finmarchicus shows a negative and positive phototactic behavior. As *Acartia* they do not show a different response for varying light intensities. The minimum threshold intensity recorded for this species is $9.9x10^{-7}$ µmol photons $m^{-2}s^{-1}$ and $9.8x10^{-6}$ µmol photons $m^{-2}s^{-1}$ for the female and juvenile (Miljeteig, Olsen et al. 2014). In their experiment, males show a positive phototactic behavior, instead females and juveniles a negative one. After the threshold level of intensity, they move in one direction reaching one side of the tank and maintain about the same distance from the light source after the next increasing of light. *Calanus* does not show variation with the different step intensity in the experiment, just they change behavior when the illumination reaches the threshold level. The interval in my experiment produces the passage up and down to the threshold intensity of light. This produces probably an on-off of reaction to the light resulting in the repeated visible migration during each round (Figure 13b), without differences in the reaction to different levels of light. In my experiment, this species has a general negative response to light exposure. This is probably because males are normally found in low quantity (Niehoff, Madsen et al. 2002).

Temora generally reacted similar to *Calanus* in my experiment. *Temora* does not show any change in swimming speed, in the dark exposure neither in the light exposure for each intensity. *Temora longicornis* shows a plasticity in and not always perform DVM. This depends on many factors such as mixing of the water, stratification, presence of food and predator (Lagadeuc, Boule' et al. 1997). This species shows a continuous movement related to the movement mouth part, and the swimming velocity is related to the presence of

predators. When copepods smell a predator they reduce the swimming rate (Van Duren and Videler 1996). The negative phototactic response is the expression of plasticity could therefore be a consequence of absence of chemical signal of food or predators.

Light level and predation risk

Copepods are the most important food base for many species of fish and their larva stages, larger zooplankton, and various others groups of animals. The light intensity plays an important role for the fish to see the prey. The ability to see prey is first of all determined by the visual sensitivity of the fish. Each species has a different light-satiation, as summarized in table 8.

Table 8: Light-satiation parameter (K_e) for three different fish species

a	*Gobiusculus flavescens*	5.0 µmol m^{-2} s^{-1}	(Aksnes and Utne 1997)
b	*Lepomis macrochirus, Pomoxis annularis*	0.05 – 0.1 µmol m^{-2} s^{-1}	(O'Brien 1987)
c	*Gadus morhua (L.), Pleuronectes platessa (L.), Scophthalmus maximus (L.)*	1.9 – 7.8 x 10^{-3} µmol m^{-2} s^{-1}	(Huse 1994)

In addition, the capacity to see prey is influenced by many other factors, like light intensity, water turbidity, as well as the size, color, movement, and the gut content of the prey. Immobile and transparent copepods produce a lower reaction distance in all light condition (Utne-Palm 1999). In our experiment the three species show a different behavior compared with the different light intensity, *Acartia tonsa* with the increasing of the light stimulus reduces the swimming speed and at 0.95 µE m^{-2} s^{-1} (third intensity level) the speed is reduced of two-thirds. *Calanus finmarchicus* reduces its swimming speed with a lower grade than *Acartia*, *Temora* does not show any difference in the swimming speed, maintains about the same speed in all conditions (Figure 16).

The slow movement for all species in the higher illumination can be interpreted as an adaptation to decrease the risk of predation. We did not find a consistent pattern with regards to size of the copepods, as the two smaller species behaved differently from each other in the

lid environment. However, coloration and visibility of copepods can be related to whether or not the gut contain food or not. A filled gut makes them more visible (Tsuda, Saito et al. 1998). In my experiment there is no food present in the water, and in addition there are no signal of predator presence (kairomones), which are also known to invoke behavioral changes (Kaartvedt 1996) (Van Gool and Ringelberg 1998). A similar environment probably left the copepods to have a risky behavior.

To evaluate how visible the three species of copepods in our experiment would be to fish predators, I calculated the range where a fish can see a prey using the following equation:

$$r = \sqrt{E'CA_p \frac{E_b}{K_e + E_b}}$$ (4) (Fiksen and MacKensie 2002)

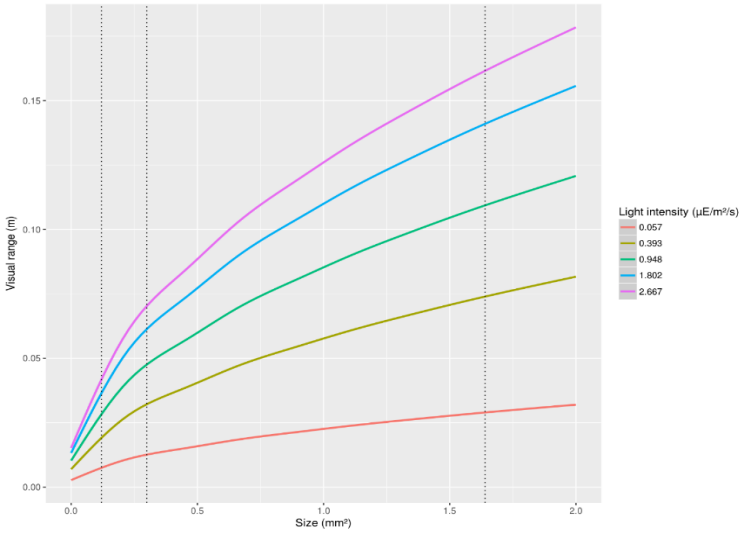
Where *E'* is the visual sensitivity of predator; *C* is the contrast of the prey; A_p is the area of the prey, which for elongate prey is equal to 0.75 * length * width (Fiksen and MacKensie 2002); E_b is the light in the environment; K_e is the light satiation of the predator. I use a contrast value of $C = 0.3$ (Utne-Palm 1999), *E'* was calculated from $T_1 = E'CA_p$ with a value of 7.5 10^{-2} µE m^{-2} s^{-1} for *Gobiusculus flavescens* and *Calanus finmarchicus* (Aksnes and Utne 1997).

In figure 18 there are three graphs of the visual range for every light intensity related to the size of the prey. The vertical dotted-lines indicate the averages sizes of the three experimental species. In figure 18a I calculated the visual distance using a light satiation parameter of K_e = 5 µE m^{-2} s^{-1} (Table 8). The visual range increases with increasing light intensity and size. While *Calanus* is the biggest species, at lower illumination the distance where it is visible for the predator becomes similar to the other two smaller species. However, at higher light intensities it is visible at a distance three times the *Acartia* distance and more than two times for *Temora*.

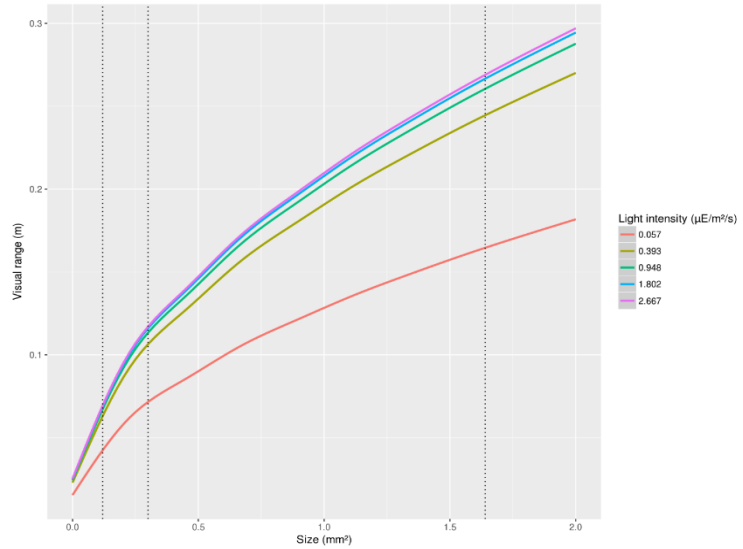
For the graph 19 b I use as K_e the higher value 0.1 µE m^{-2} s^{-1} to simulate a very light sensitive predator species. It is clear how the lower value of the parameter K_e reduces the difference for the visual range at each intensity. Predators with this higher sensitivity can detect at longer distance the prey. The low illumination value is below the light saturation threshold K_e of the predator, this maintains a different visual range for the three species, but other values are over the parameter and the visual range become for each intensity very similar.

In the last graph c the K_e value is 0.005 µE m^{-2} s^{-1} took from the range values. All five light intensities used in our experiment exceed this low light saturation constant K_e and the difference in visual range is only driven by the size of the copepods. These small exercise illustrations how the light environment and the light sensitivity are important factors for predators to see their prey. It is easy to expect stronger light reactions for the biggest species *Calanus* or a modification in swimming speed for the colored *Temora*. The smaller and transparent species *Acartia* has the stronger light reaction despite predators can detect closer the other two species.

A



B



C

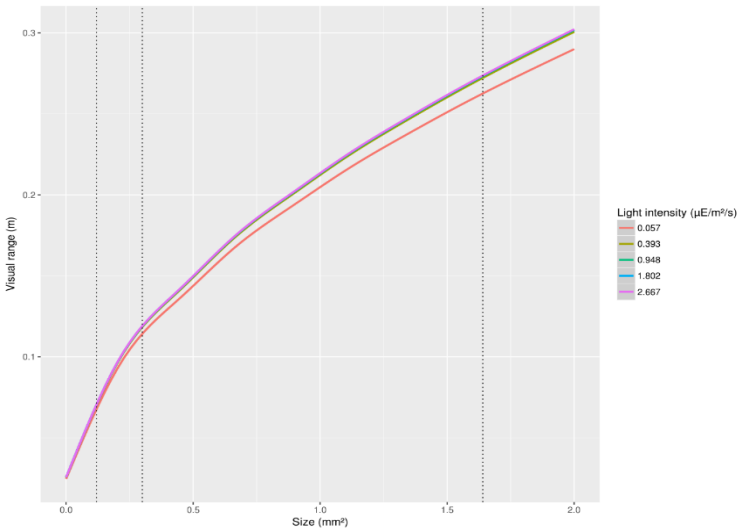


Figure 19: Model estimate of visual range for fish five light intensity as a function of size. The vertical lines represent the three species of copepods used in my experiment. From the left: Acartia = 0.12 mm^2; Temora = 0.30 mm^2; Calanus 1.64 mm^2

Conclusion

The three copepods species in my experiment reacted differently when exposed to light. While *Temora* and *Calanus* showed constant swimming speeds at all light levels, *Acartia* increased speed at the lowest light level. The perceived light intensity seems not only to depend on the light intensity and species but also on the number of light exposures. My results suggest that light, once above a certain threshold intensity, triggers copepods to change position in the water. The direction depended on the species. The small *Acartia* swam towards the light, while the more visible large *Calanus* and pigmented *Temora* escaped from the light.

Low light intensity reduces the visual range of fish predators (Aksnes, et al. 2004). I therefore expected a stronger avoidance reaction of the bigger and more pigmented species. My results suggest that size or contrast of copepods interact with the pattern of light exposure to explain motility behavior

I only used the light intensity as an independent factor in my experiment. However behavior is the result of the interaction between many factors, such as presence of food, predator, and the internal state of the animal (e.g. starvation). There are many possible directions for new experiments, for example testing the effects of different light spectra, or of continuous variations in light intensity. My setup allows for easy manipulating the spectral composition and the intensity of the light. it is thus particularly suited to address the above questions, either with focus on light or in combination with other environmental drivers.

Reference

Aarseth, K. A. and S. T. A. (1999). "Wavelength-specific behavior in Lepeophtherios salmonis and Calanus finmarchicus to ultraviolet and visible light in laboratory experiments (Crustacea: Copepoda)." Mar Ecol Prog Ser **186**: 211-217.

Aksnes, D. L. and A. C. W. Utne (1997). "A revised model of visual range in fish." Sarsia **82(2)**: 137-147.

Barrientos, Y. (1980). Ultrastructure of sensory units of the first antennae of calanoid copepods. M.Sc., University of Ottawa, ON.

Boxshall, G. A. (1992). Copepoda. in "Microscopy Anatomy of Invertebrates".

Buskey, E. J. (1994). "Factor affectin feeding selectivity of visual predatos on the copepod *Acartia tonsa*: locomotion, visibility and escape responses." Hydrologia **292/293**: 447-453.

Buskey, E. J. and D. Hartline (2003). "High-speed video analysis of the escape responses of the copepods *Acartia tonsa* to shadows." Biol. Bull. US **204**: 28-37.

Champalbert, G. (1978). "Rythines d'activite natatoire de quelques copepodes hyponeustoniques (*Anomulocera patersoni* Templeton, *Pontella mediterranea* Claus, *Labidocera wollastoni* Lubbock) en fonction des conditions d'eclairement et de pression." Journal of Experimental Marine Biology and Ecology **35**(233-239).

Champalbert, G. (1979). "Rythme d'activite natatoire de deux especes de copepodes pontellides (*Anornaloceru putersoni et Pontella rnediterranea*)." Tethys **9**: 83-85.

Cohen, J. H. and R. B. J. Forward (2002). "Spectral Sensitivity of Vertically Migrating Marine Copepods." Biol. Bull. **203**: 307-314.

Cohen, J. H. and R. B. J. Forward (2005). "Photobehavior as an inducible defence in the marine copepods Calanopia americana." Limnol Oceanogr. **50**(4): 1296-1277.

Dagg, M. J., et al. (1997). "Vertical migration and feeding behavior of Calanus pacificus females during a phytoplankton bloom in Dabob Bay, U.S." Limnol. Oceanogr. **42**(5): 974-980.

Fields, D. M., et al. (2002). "Mechanical and neural responses from the mechanosensory hairs on the antennule of *Gaussia princeps*." Mar Ecol Prog Ser **227**: 173-186.

Fiksen, Ø. and B. R. MacKensie (2002). "Process-based models of feeding and prey selection in larval fish." Mar Ecol Prog Ser **243**: 151-164.

Hallberg, E., et al. (1997). "Olfactory sensilla in crustaceans: morphology, sezual dimorphism, and distribution patterns." Int. F. Insect Morphol. Embryol. **26**: 173-180.

Hardy, A. C. and R. Bainbridge (1954). "Experimental obseravations on the vertical migration of planktonic animals." Journal of the Marine Biological Association of the United Kingdom **33**: 409-448.

Hay, S. J., et al. (1991). " Zooplankton biomass and production in the North Sea during the autumn circulation experiment." Continental Shelf Research **11**: 1453-1476.

Heuschele, J. and E. Selander (2014). "The chemical ecology of copepods." Journal of Plankton Research **36**(4): 895-913.

Huse, I. (1994). "Feeding at different illumination levels in larvae of 3 marine teleost species: cod, *Gadus morhua* L., plaice, *Pleuronectes platessa* L., and turbot, *Scophthalmus maximus* (L.).
." Aqua Fish Man **25**: 687-695.

Huys, R. and G. A. Boxshall (1991). Copepod Evolution.

Kaartvedt, S. (1996). "Effects of copepod foranging behaviour on predation risk: an experimental study of the predatory copepod Paraeuchaeta norvegica feeding on Acartia clausi and A. tonsa (copepoda)." Limnol Oceanogr. **42**(1): 164.

Kiørboe, T. and A. W. Visser (1999). "Predator and prey perception in copepods due to hydromechanical signals " Mar Ecol Prog Ser **179**: 81-95.

Lagadeuc, Y., et al. (1997). "Effect of vertical mixing on the vertical distribution of copepods in costal waters." Journal of Plankton Research **19**(9): 1183-1204.

Maszczyk, P. (2016). "Miniature plankton columns to study the depth distribution of zooplankton at gradients of food, predation risk, temperature, and UV radiation." limnol Oceanogr. Methods **14**(3): 210–223

Mauchline, J. (1998). The Biology of Calanoid Copepods.

Miljeteig, C., et al. (2014). "Sex and life stage dependent phototactic response of the marine copepod Calanus finmarchicus (Copepoda: Calanoida)." Journal of Experimental Marine Biology and Ecology **451**: 16-24.

Niehoff, B., et al. (2002). "Reproductive cycles of three dominant Calanus species in Disko Bay, West Greenland." Mar. Biol. **140**(3): 567-576.

Nielsen, T. G. and P. J. Hansen (1999). Dyreplankton i danske farvande.

O'Brien, W. J. (1987). "Planktivory by freshwater fish: thrust and parry in the pelagia. In: Kerfoot WC, Sih A(eds) Predation: direct and indirect impacts on aquatic communities." University Press of New England, London: 3-16.

Paffenofer, G. A. and P. A. Loyd (1999). "Ultrastructure of setae of the maxilliped of the marine planktonic copepod *Temora stylifera*." Mar Ecol Prog Ser **178**: 101-107.

Petzoldt, T., et al. (2009). "Effects of zooplankton diel vertical migration on a phytoplankton community: A scenario analysis of the underlying mechanisms." Ecologial Modelling **220**(9-10): 1358-1368.

Sokal, R. R. and F. J. Rohlf (1995). Biometry: the principles and practice of statistics in biological research, W.H. Freeman, New York.

Stearns, D. E. (1986). "Copepod grazing behavior in simulated natural light and its relation to nocturnal feeding." Mar Ecol Prog Ser **30**: 63-76.

Stearns, D. E. and R. B. J. Forward (1984 a). "Photosentitivity of the calanoid copepod Acartia tonsa." marine biology **82**(1): 85-89.

Stearns, D. E. and R. B. J. Forward (1984 b). "Copepod photobehavior in a simulated natural light environment and its relation to nocturnal vertical migration." marine biology **82**(1): 91-100.

Svetlichny, L. S. and B. V. Kurbatov (1983). "Effect of the body sizes on spatial-time and energy parameters ol the vertical migration of copepods." Ekologiya Morya **27**: 78-85.

Svetlichny, L. S. and I. Y. Yarkina (1989). "Locomotion rhythm in *Calanus helgolandicus* (Crustacea, Copepoda)." Zoologichesky Zhurnal **68**(6): 50-55.

Tiselius, P. and P. R. Jonsson (1990). "Foraging behaviour of six calanoid copepods: observations and hydrodynamic analysis." Mar Ecol Prog Ser **66**: 23-33.

Titelman, J. and T. Kiørboe (2003). "Motility of copepod nauplii and implications for food encounter." Mar Ecol Prog Ser **247**: 123-135

Tranter, D. J., et al. (1981). "Nocturnal movementes of phototactic zooplankton in shallow waters." marine biology **61**: 317-326.

Tsuda, A., et al. (1998). "Effect of gut content on the vulnerability of copepods to visual predation." Limnol Oceanogr. **43**(8): 1944-1947.

Utne-Palm, A. C. (1999). "The effect of prey mobility, prey contrast, turbidity and spectral composition on the reaction distance of Gobiusculus flavescens to its planktonic prey." Journal of Fish Biology **54**: 1244-1258.

Van Duren, L. A. and J. J. Videler (1996). "The trade-off between feeding,mate seeking and predator avoidance in copepods: behavioural responses to chemical cues." Journal of Plankton Research **18**(5): 805-818.

Van Gool, E. and J. Ringelberg (1998). "Light-induced migration behaviour of *Daphnia* modified by food and predator kairomones." Animal behavior **56**(3): 741-747.

Waggett, R. J. and E. J. Buskey (2007). "Calanoid copepod escape behavior in response to a visual predator." marine biology **150**: 599-607.

Wikipedia (2017). "List of refractive indices." from https://en.wikipedia.org/wiki/List_of_refractive_indices.

Appendix

Standard Operating Procedure

1. During the first run it is useful to fill the cylindrical tank with simple tap water, place it in the glass tank and fill it until the cylindrical tank is totally submerged. Now remove the tank and make a mark on the water level in the glass tank.
2. Prepare the necessary amount of filtered seawater at 30 psu and keep it in the experimental room for at least one day so the water has the same temperature.
3. Collect the animals for the experiment from the stock culture using a pipette.
4. Divide the animals to at 6-well plate, in groups of 5 individuals per well.
5. Add filtered seawater to each well to give them enough space to swim.
6. Place the plate in the experimental room to maintain the correct temperature.
7. Switch on the Raspberry. It is important control that the external hd is mounted in the right path.
8. Control the alignment of camera with a preview script. It is good idea to use an indication mark on the tank to ensure the same alignment.
9. Clean the glass tank with alcohol 75%.
10. Fill glass tank with filtered water until the mark made in point 1.
11. Clean the external of the tank with paper and alcohol 75% if necessary, be gentle as the acrylic is a delicate material.
12. Insert the black plastic tube holders if not already present.
13. Fill the cylindrical tank with the temperate filtered seawater up until 0.5 cm to the top.
14. Remove any bubbles attached to the surface of the tank with a small brush.
15. Take, with a pipette modified with a large hole, one group of copepods from the plate and gently insert them into the cylindrical tank.
16. Add some more filtered seawater, until there is about 2 mm space left to the top of the tank.
17. Close the tank with the lid. Incline the lid to on one side and start to insert the lid. This way the air can come out only on one side, and at the same time it is possible to control that no animal is flushed out.
18. Bring the cylindrical tank and place in the glass tank using the handles on the holders, again starting on one side. Do not be in a rush as it can create waves inside the glass tank.
19. Move and align the cylindrical tank close to the light source.
20. Turn on the IR light (if it is the first group), and confirm that it works by observing a faint red light come out from the LED.
21. Close the glass tank with the acrylic light diffusor and the IR light strip.
22. Close the box with the black plastic cover.

23. Open the file light.dat on the raspberry and insert the sequence of intensity, save and exit.
24. Control that all lights in the room are switched off.
25. Start the python script.
26. Take note of day and time for each replicate. It useful to check that everything is recorded in right way.
27. Exit the room and close the door.
28. Calculate when the 5 sequential exposure rounds are finished.
29. Come back few minutes before to prepare the second tank as in points 11-14.
30. When the program finished, enter the room and take another group from the plate.
31. Repeat the points 15, 16 and 17.
32. Remove the glass tank light diffusor and remove slowly the tank.
33. Remain a few second over the glass tank to left drops down a few water
34. Take care to do not left drop water over the LED or other electronic part!
35. Bring out the previous tank and put aside.
36. Repeat the procedure from point 18 to 28.
37. Put the tank on a table or a flat surface in vertical position.
38. Gently open the tank.
39. To collect the copepods can use different technic, it depends by the species
    - For *Temora* and *Calanus* the pipette and a small beaker are enough, as they are easy to sea in the tube.
    - *Acartia* is more difficult, as they are small and fast. Use a light source to bring them to the surface and now can collect with the pipe or the modified pipette
40. Transfer the copepod to the glass slide to take a photo.
41. When take the photo control that all animals are visible from dorsal side.
42. Rename the photos the number of the species and group.
43. After that you can transfer the copepod to a beaker to keep them alive or freeze them in a 1.5 mm Eppendorf tube, on the tube write species and number of group
44. Repeat from the point 29 until finish the group in the plate.
45. After the last group, it is necessary to wash the cylindrical tank, glass tank and other tools with fresh water.
46. Put aside and let dry.
47. Switch off the Raspberry pi and disconnect the power.

Arduino code

This code is uploaded on the Arduino board to control the NeoPixel LED.

```
#include <Adafruit_NeoPixel.h>
#ifdef __AVR__
#include <avr/power.h>
#endif

#define PIN            6 //pin for neopixel data
#define NUMPIXELS      1// number of pixels

int rgb_count=0;
int a;
String R;
String G;
String B;
String leds;
String duration;
String wait;
String xvalue;
int R_int=0;
int G_int=0;
int B_int=0;
boolean on_light= false;

//assign to pixels the setting for the board
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS, PIN,
NEO_GRB + NEO_KHZ800);

void setup() {
  Serial.begin(9600);
  pixels.begin();
  pixels.show();

}

void loop(){
  if (Serial.available())
  {
    //Serial.print(Serial.readString());
    xvalue = Serial.readStringUntil(',');
    Serial.print("valore da rasp: ");
    Serial.println(xvalue);

    switch (rgb_count){
    case 0:
      //Serial.print("xvalue: ");
      //Serial.println(xvalue);
```

```
      R=xvalue;
      rgb_count=rgb_count+1;
   //   Serial.print("valroe di R: ");
      //Serial.println(R);
      break;
    case 1:
      G=xvalue;
      rgb_count++;
//      Serial.print("valroe di G: ");
      //Serial.println(G);

      break;
    case 2:
      B=xvalue;
//      Serial.print("valroe di B: ");
      //Serial.println(B);
      rgb_count++;
      break;
    case 3:
      leds=xvalue;
      //Serial.println(leds);
      rgb_count++;
      break;
    case 4:
      wait=xvalue;
      rgb_count++;
      break;
    case 5:
      duration=xvalue;
      rgb_count=0;
      on_light=true;
      //Serial.println(duration);
      break;
    }
    R_int=R.toInt();
    G_int=G.toInt();
    B_int=B.toInt();


  }
  if (on_light){

    delay(wait.toInt());
 //   Serial.println(wait.toInt());
    //    Serial.print("led: ");
    //    Serial.print(leds.toInt());
    Serial.print("duration: ");
    Serial.println(duration.toInt());
```

```
    on_light=neopixel_start (leds.toInt(), duration.toInt(),
R_int, G_int, B_int);

  }


}

//function to turn on or off the light with a fade in
boolean neopixel_start (int posit,long time_on,float red,
float green, float blue) {


  // fade in
  float redi;
  float greeni;
  float bluei;
    for (int i=0; i<200; i++) {
      redi= red/200;
      //Serial.println(redi);
      greeni=(green/200)*i;
      //Serial.println(greeni);
      bluei=(blue/200)*i;
      //Serial.println(bluei);
      pixels.setPixelColor(posit, redi*i, greeni, bluei);
    //  pixels.setPixelColor(posit+1, redi*i, greeni, bluei);
     // pixels.setPixelColor(posit+8, redi*i, greeni, bluei);
     // pixels.setPixelColor(posit+9, redi*i, greeni, bluei);
      pixels.show();
      delay(10);

  }

  // luce piena per tempo rimanente
  //Serial.println("accendo i led");
  pixels.setPixelColor(posit, red, green, blue);
  //pixels.setPixelColor(posit+1, red, green, blue);
  //pixels.setPixelColor(posit+8, red, green, blue);
  //pixels.setPixelColor(posit+9, red, green, blue);
  pixels.show();

  Serial.println(time_on);
  delay(time_on);

  pixels.setPixelColor(posit,0,0,0);
  //pixels.setPixelColor(posit+1,0,0,0);
  //pixels.setPixelColor(posit+8,0,0,0);
  //pixels.setPixelColor(posit+9,0,0,0);
  //Serial.println("passato dall'azzerrare");
```

```
  pixels.show();
  Serial.println("luce spenta");

  boolean result=false;
  return result;

}
```

Raspberry Pi 3 code

This code is on the Raspberry Pi 3 to control the camera and to send to Arduino the intensity of the single colour light. The value is read from the file light.dat. An example of the file further.

```
import random
import serial
import time
import picamera
import shutil
from threading import Thread
import os
import sys


camera = picamera.PiCamera()

camera.resolution = (1600,600)
camera.iso = 200
camera.color_effects =(128,128)
camera.framerate = 25

print "15 min in dark"
time.sleep(900)
print "finish adaptation, start recording"

def ssh_function(tempo, nomefile):
        comando="raspivid -fps 25 -ISO 200 -w 1600 -h 600 -cfx
128:128 -t "+str(tempo)+" -o
/media/pi/legoblack/video_thesis_black/"+nomefile
        print comando
     data = {"user":"pi",
          "host":"192.168.0.3",
          "password":"raspberry",
          "command": comando}
          #"command": "raspivid -fps 25 -ISO 100 -w 1600 -h
500 -t "+str(tempo)+" -o
/media/pi/legoblack/video_thesis_black/"+nomefile}

     command ="sshpass -p {password} ssh -x {user}@{host}
{command}"
     print "send comand via ssh"
     os.system(command.format(**data))
     print "pic took"

#camera.start_preview()
ser = serial.Serial(
     port='/dev/ttyUSB0',
```

```
     baudrate= 9600)

#open file for reading plus possibility to write
light_file=open ("/home/pi/thesis/light.dat", "r")
print("light.dat opened ")
valore_linea = light_file.readlines() #letto numero linee,
valore string
num_linee = int(len(valore_linea)) #numero linee in variabile
int
light_file.close()

for x in range(num_linee):

    #open file with values
    light_file=open ("/home/pi/thesis/light.dat", "r")

    valore_linea = light_file.readlines()
    valore = valore_linea[x]
    light_file.close()

    #divide values string
    valore_lista=valore.split()

    #send values to arduino
    ser.write(str(valore_lista[0]))
    print(valore_lista[0])
    ser.write(",")
    ser.write(str(valore_lista[1]))
    print(valore_lista[1])
    ser.write(",")
    ser.write(str(valore_lista[2]))
    print(valore_lista[2])
    ser.write(",")
    ser.write(str(valore_lista[3]))
    print(valore_lista[3])
    ser.write(",")
    ser.write(str(valore_lista[4]))
    print(valore_lista[4])
    ser.write(",")
    ser.write(str(valore_lista[5]))
    print(valore_lista[5])
    ser.write(",")

    #path="/media/pi/gnatos/media_thesis/"
    #momento=time.strftime("%d%m%y_%H%M", time.gmtime())
    #nome_compl=path+momento+".h264"

#print("/media/pi/gnatos/media_thesis/"+time.strftime("%d%m%y_
%H%M", time.gmtime())+".h264")
```

```
    #print (nome_compl)

    #start recording video with data and time as name for
total time

camera.start_recording("/media/pi/gnatos/video_pietro/"+time.s
trftime("%d%m%y_%H%M",
time.gmtime())+str(x)+".h264")#,quality=1)
    nomefile=time.strftime("%d%m%y_%H%M",
time.gmtime())+str(x)+"_black.h264"
    tempofile=(int(valore_lista[4])+int(valore_lista[5]))+2000
    print tempofile
    thethread= Thread(target=ssh_function(tempofile,nomefile))
    thethread.start()

#time.sleep(((int(valore_lista[4])+int(valore_lista[5]))/1000)
+2)
    print "thread started"
    #time.sleep(tempofile/1000)
    camera.stop_recording()
    #time.sleep(5)
#camera.stop_preview()
```

Python tracking code

Script in Python to analyse the video recorded and extract the position and other parameters.

```
# import the necessary packages
import numpy as np
import argparse
import time
import cv2
import os
import sys
import wx


# Setup SimpleBlobDetector parameters.
params = cv2.SimpleBlobDetector_Params()


maxpoints = None  # maximum number of blobs to track at once
vfilt = None  # stepwise velocity filter factor (low-pass
filter)
maxvel = None  # maximum physically likely velocity (delta-X
per frame)
xdistThresh = None # how many pixels must a blob travel,
before it becomes an event?

global xc
global yc
global firstencounter
global xo
global yo
global grossdisto
global yvel
global xvel
global realvel
global xvelFilt
global xstart
global ystart
global grossdist
global xdist
global ydist
global tracktime
global tracknr
global tracktimeo
global inver_frame
global font
global directory
directory=["/home/gnatophillum/ownCloud/Documents/oslo/UiO/the
sis/python_arduino/tracking/",""]
```

```
os.chdir(directory[0])
#os.chdir("/home/gnatophillum/ownCloud/Documents/oslo/UiO/thes
is/python_arduino/tracking/")
# ----------------------------------------------------


def load_setting():
  load_file=open("setup.dat","r")
  print "file aperto"
  load_value=load_file.readlines()
  print load_value
  values=load_value[0].split()

  #use global variables
  global maxjump
  global history
  global ratiomin
  global ratiomax
  global varThreshold
  global bShadowDetection
  global maxpoints  # maximum number of blobs to track at once
  global vfilt   # stepwise velocity filter factor (low-pass
filter)
  global maxvel  # maximum physically likely velocity (delta-X
per frame)
  global xdistThresh

  #fill the variables
  maxjump=values[0]
  ratiomin=values[1]
  ratiomax=values[2]
  params.minThreshold=float(values[3])
  params.maxThreshold=float(values[4])
  params.minDistBetweenBlobs=float(values[5])
  params.filterByArea=values[6]
  params.minArea=float(values[7])
  params.maxArea=float(values[8])
  params.filterByCircularity=values[9]
  params.minCircularity=float(values[10])
  params.filterByConvexity=values[11]
  params.minConvexity=float(values[12])
  params.filterByInertia=values[13]
  params.minInertiaRatio=float(values[14])
  history=values[15]
  varThreshold=values[16]
  bShadowDetection=values[17]
  maxpoints =  values[18]
  vfilt = values[19]
```

```
  maxvel = values[20]
  xdistThresh = values[21]
  load_file.close()


# ------------------------------------------------


def
tracking(thename,moviename):#,l_history,l_varThreshold,l_bShad
owDetection):

  global history
  global varThreshold
  global detectShadows
  global maxpoints  # maximum number of blobs to track at once
  global vfilt   # stepwise velocity filter factor (low-pass
filter)
  global maxvel  # maximum physically likely velocity (delta-X
per frame)
  global xdistThresh

  print "primo step funzione tracking"

  kernel5 = np.array([[0, 1, 1, 1, 0],
                      [1, 1, 1, 1, 1],
                      [1, 1, 1, 1, 1],
                      [1, 1, 1, 1, 1],
                      [0, 1, 1, 1, 0]]).astype(np.uint8)

  kernel3 = np.array([[0, 1, 0],
                      [1, 1, 1],
                      [1, 1, 0]]).astype(np.uint8)

  #maxpoints = 5  # maximum number of blobs to track at once
  #vfilt = 0.2  # stepwise velocity filter factor (low-pass
filter)
  #maxvel = 1000  # maximum physically likely velocity (delta-
X per frame)
  #xdistThresh = 2  # how many pixels must a blob travel,
before it becomes an event? #ex40
  xc = np.float32(np.zeros(maxpoints))  # x center coordinates
  yc = np.float32(np.zeros(maxpoints))  # y center coordinates
  xo = np.float32(np.zeros(maxpoints))  # x center, previous
frame
  yo = np.float32(np.zeros(maxpoints))  # x center, previous
frame
  grossdisto = np.float32(np.zeros(maxpoints))  #
grossdistancecounter
```

```
  yvel = np.float32(np.zeros(maxpoints))  # y velocity,
instantaneous
  xvel = np.float32(np.zeros(maxpoints))  # x velocity,
instantaneous
  realvel = np.float32(np.zeros(maxpoints))  # real velocity,
instantaneous
  firstencounter = np.float32(np.zeros(maxpoints))
  xvelFilt = np.float32(np.zeros(maxpoints))  # x velocity
(filtered by rolling average)
  xstart = np.float32(np.zeros(maxpoints))  # x starting point
(for distance-travelled)
  ystart = np.float32(np.zeros(maxpoints))  # x starting point
(for distance-travelled)
  grossdist = np.float32(np.zeros(maxpoints))  #grossdistance
travelled since starting point
  xdist = np.float32(np.zeros(maxpoints))  # x distance-
travelled since starting point
  ydist = np.float32(np.zeros(maxpoints))  # y distance-
travelled since starting point
  tracktime = np.float32(np.zeros(maxpoints))
  trackнr = np.float32(np.zeros(maxpoints))
  tracktimeo = np.float32(np.zeros(maxpoints))
  font = cv2.FONT_HERSHEY_SIMPLEX  # for drawing text


  # define the upper and lower boundaries for a color
  # to be considered "blue"
  #blueLower = np.array([0, 74, 94])
  #blueUpper = np.array([14, 255, 172])


  blueLower = np.array([140], dtype="uint8")#140
  blueUpper = np.array([255], dtype="uint8")

  # loop through all the movies

#os.chdir("/home/gnatophillum/ownCloud/Documents/oslo/UiO/thes
is/video/02_03/")
  # ----------------------------------------------------



  ## Create a detector with the parameters
  detector = cv2.SimpleBlobDetector_create(params) #before was
SimpleBlobDetector_create

  print "secondo step funzione tracking"

  # ----------------------------------------------------
```

```
  if moviename.endswith(".h264") or moviename.endswith(".mp4")
or moviename.endswith(".mkv"):

       # ------------------------------------------------------

       fgbg =
cv2.createBackgroundSubtractorMOG2(history,varThreshold,detect
Shadows=True)#=800, varThreshold=5,
detectShadows=True)#bShadowDetection=True)
       # ------------------------------------------------------

       camera = cv2.VideoCapture(thename)
       #camera.set(cv2.CAP_PROP_FPS, 1)

       # initalizing frame counter
       frame_count = 0

       f = open(str(thename) + "_trackdata.csv", "w")
       names = str("Movie, Frame, X, Y, velocity, distance,
particlenr, tracknr, tracktime \n")
       f.write(names)
       #f.close()
       print("scritto file")
       firstencounter[:] = 1
       tracknr[:] = 1
       xo[:] = 0
       yo[:] = 0
       # keep looping
       while True:
           # grab the current frame
           (grabbed, frame) = camera.read()

           # Updating the framecounter
           frame_count += 1

           # check to see if we have reached the end of the
           # video
           if not grabbed:
            break

           # This crops the movie to the desired area, here it
is the maze. (y1:y2, x1,x2)
           #frame = frame[270:720, 284:884]
           fgmask = fgbg.apply(frame)

           temp = cv2.inRange(fgmask, blueLower, blueUpper)
```

```
            temp2 = cv2.erode(temp, kernel3, iterations=2)  #
remove isolated noise pixels with small kernel
            filtered = cv2.dilate(temp2, kernel5, iterations=3)

            filtered = cv2.dilate(temp2, kernel5, iterations=2)
            #filtered = cv2.blur(filtered, (3, 3))

            inv = 255 - filtered  # invert black to white
            #inv = 255 - fgmask  # invert black to white

            # Detect blobs
            keypoints = detector.detect(inv)
            i = 0

            if len(keypoints) > 0:

             for k in keypoints:
                 if (i < maxpoints):
                  xc[i] = k.pt[0]  # x center of blob
                  yc[i] = k.pt[1]  # y center of blob
                  xs1 = int(k.pt[0])  # integer coords
                  ys1 = int(k.pt[1])
                  radius = int(k.size / 2)
                  xvel[i] = xc[i] - xo[i]  # delta-x since
previous frame
                  yvel[i] = yc[i] - yo[i]  # delta-y since
previous frame
                  realvel[i] = ((xvel[i])**2+(yvel[i]**2))**0.5
# real velocity, instantaneous

                  if firstencounter[i] > 0 or abs(xvel[i]) >
maxjump or abs(yvel[i]) > maxjump:
                      realvel[i] = 0
                      grossdist[i] = 0
                      firstencounter[i] = 0
                      tracktime[i] = 0  # tracktime is a
counter which gets reset everytime the particle is lost, can
be used to filter the dataset later
                      tracknr[i] += 1

                  xdist[i] = xc[i] - xstart[i]  # calculate x-
distance this blob has travelled so far
                  ydist[i] = yc[i] - ystart[i]  # calculate y-
distance this blob has travelled so far

                  grossdist[i] = grossdisto[i] + realvel[i]  #
real velocity, instantaneous
                  tracktime[i] += 1
```

```
                newdata = (str(moviename) + "," +
str(frame_count) + "," + str(xc[i]) + "," + str(yc[i]) + "," +
str(realvel[i]) + "," + str(grossdist[i]) + "," + str(i) +
"," + str(tracknr[i])+ "," + str(tracktime[i])+ "\n")
                f.write(newdata)
                new = cv2.circle(frame, (xs1, ys1), radius,
[0, 50, 255], 2)
                new2 = cv2.circle(fgmask, (xs1, ys1), radius,
[0, 50, 255], 2)
                #new3 = cv2.putText(frame, str("x:
"+str(xs1)+"y: "+str(ys1)), (int(xs1), int(ys1)),
cv2.FONT_HERSHEY_SIMPLEX, 1, (255,0,0),1)
                #im = cv2.circle(fgmask, (xs1, ys1), radius,
[0, 50, 255], 2)
                xo[i] = xc[i]
                yo[i] = yc[i]# remember current x-center
value for next frame
                grossdisto[i] = grossdist[i]
                i += 1

           else:
            #newdata = (str(moviename) + "," +
str(frame_count) + ",NA,NA,NA,NA,NA,NA,NA\n")
            firstencounter[:] = 1
            #f.write(newdata)

       # show the frame and the binary image
           cv2.imshow(thename, cv2.resize(frame, (1000, 250)))
           cv2.imshow("inverted", cv2.resize(inv, (1000,
250)))
           cv2.imshow("grayscale", cv2.resize(fgmask, (1000,
250)))
           #time.sleep(0.5)




  # if the 'q' key is pressed, stop the loop
           if cv2.waitKey(1) & 0xFF == ord("q"):
            break

      # cleanup the camera and close any open windows
       f.closed
       camera.release()
       cv2.destroyAllWindows()


load_setting()
```

```
 #------------------------------------------------------------
----------
 # GUI setting and functions
class MyForm(wx.Frame):

    #------------------------------------------------------------
-------------
    def __init__(self):
        wx.Frame.__init__(self, None, wx.ID_ANY, "Open CV
Tracking", size=(500,570))

        # Add a panel so it looks the correct on all platforms
        panel = wx.Panel(self, wx.ID_ANY)


        #add menubar and menu item
        menubar = wx.MenuBar()
        menu_item= wx.Menu()
        menu_item2= wx.Menu()
     menubar.Append(menu_item,"File")
     #menubar.Append(menu_item2,"?")
     apri = menu_item.Append(-1, "Open")
     setting = menu_item.Append(-1, "Save setting")
     esci = menu_item.Append(-1, "Quit")
     #info = menu_item2.Append(-1, "Manual")

     self.SetMenuBar(menubar)
     #set event for click on menu items
     self.Bind(wx.EVT_MENU, self.on_clic_apri, apri)
     self.Bind(wx.EVT_MENU, self.on_clic_setting, setting)
     self.Bind(wx.EVT_MENU, self.on_clic_esci, esci)
     #self.Bind(wx.EVT_MENU, self.on_clic_info, info)

     load_setting()

     #checkbox and value in the panel
     self.maxj = wx.SpinCtrl(self, -1, str(maxjump), (140,
30), (60, -1), min=0, max=500)
     self.ratmi = wx.SpinCtrlDouble(self, -1, str(ratiomin),
(140, 60), (60, -1), min=0.0, max=10.0, inc=0.01)
     self.ratma = wx.SpinCtrlDouble(self, -1, str(ratiomax),
(140, 90), (60, -1), min=0.0, max=10.0, inc=0.1)
     self.minth = wx.SpinCtrl(self, -1,
str(params.minThreshold), (140, 120), (60, -1), min=0,
max=200)
     self.maxth = wx.SpinCtrl(self, -1,
str(params.maxThreshold), (140, 150), (60, -1), min=0,
max=500)
```

```
     self.distm = wx.SpinCtrl(self, -1,
str(params.minDistBetweenBlobs), (140, 180), (60, -1), min=0,
max=500)
     self.areaf=wx.CheckBox(self, -1,'', (100,210))
     self.areami = wx.SpinCtrl(self, -1, str(params.minArea),
(140, 210), (60, -1), min=0, max=5000)
     self.areama = wx.SpinCtrl(self, -1, str(params.maxArea),
(300, 210), (60, -1), min=0, max=5000)
     self.circu=wx.CheckBox(self, -1,'', (100,240))
     self.circum = wx.SpinCtrlDouble(self, -1,
str(params.minCircularity), (140, 240), (60, -1), min=0,
max=5, inc=0.1)
     self.conv=wx.CheckBox(self, -1,'', (100,270))
     self.convm = wx.SpinCtrlDouble(self, -1,
str(params.minConvexity), (140, 270), (60, -1), min=0, max=5,
inc=0.01)
     self.inertia=wx.CheckBox(self, -1,'', (100,300))
     self.inertiam = wx.SpinCtrlDouble(self, -1,
str(params.minInertiaRatio), (140, 300), (60, -1), min=0,
max=5, inc=0.01)
     self.histo = wx.SpinCtrl(self, -1, str(history), (140,
330), (60, -1), min=10, max=1000)
     self.varthre = wx.SpinCtrl(self, -1, str(varThreshold),
(140, 360), (60, -1), min=0, max=50)
     self.shadowd=wx.CheckBox(self, -1,'', (100,390))
     self.maxpoint = wx.SpinCtrl(self, -1, str(maxpoints),
(140, 410), (60, -1), min=0, max=50)
     self.vfilt = wx.SpinCtrlDouble(self, -1, str(vfilt),
(140, 440), (60, -1), min=0, max=10, inc=0.01)
     self.maxvel = wx.SpinCtrl(self, -1, str(maxvel), (140,
470), (60, -1), min=100, max=2000)
     self.xdistthre = wx.SpinCtrl(self, -1, str(xdistThresh),
(140, 500), (60, -1), min=0, max=100)


     #event of change
     self.maxj.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.ratmi.Bind(wx.EVT_SPINCTRLDOUBLE, self.on_changed)
     self.ratma.Bind(wx.EVT_SPINCTRLDOUBLE, self.on_changed)
     self.minth.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.maxth.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.areaf.Bind(wx.EVT_CHECKBOX, self.on_changed)
     self.areami.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.areama.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.circu.Bind(wx.EVT_CHECKBOX, self.on_changed)
     self.circum.Bind(wx.EVT_SPINCTRLDOUBLE, self.on_changed)
     self.conv.Bind(wx.EVT_CHECKBOX, self.on_changed)
     self.convm.Bind(wx.EVT_SPINCTRLDOUBLE, self.on_changed)
     self.inertia.Bind(wx.EVT_CHECKBOX, self.on_changed)
```

```
     self.inertiam.Bind(wx.EVT_SPINCTRLDOUBLE,
self.on_changed)
     self.histo.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.varthre.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.shadowd.Bind(wx.EVT_CHECKBOX, self.on_changed)
     self.maxpoint.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.vfilt.Bind(wx.EVT_SPINCTRLDOUBLE, self.on_changed)
     self.maxvel.Bind(wx.EVT_SPINCTRL, self.on_changed)
     self.xdistthre.Bind(wx.EVT_SPINCTRL, self.on_changed)

     #description
     wx.StaticText(self, -1, 'Options', (20,10))
     wx.StaticText(self, -1, 'Maxjump', (10,30))
     wx.StaticText(self, -1, 'Ratiomin', (10,60))
     wx.StaticText(self, -1, 'Ratiomax', (10,90))
     wx.StaticText(self, -1, 'Min threshold', (10,120))
     wx.StaticText(self, -1, 'Max threshold', (10,150))
     wx.StaticText(self, -1, 'Min dist blobs', (10,180))
     wx.StaticText(self, -1, 'Area', (10,210))
     wx.StaticText(self, -1, 'Area min', (210,210))
     wx.StaticText(self, -1, 'Area max', (370,210))
     wx.StaticText(self, -1, 'Circularity', (10,240))
     wx.StaticText(self, -1, 'Convexity', (10,270))
     wx.StaticText(self, -1, 'Inertia', (10,300))
     wx.StaticText(self, -1, 'History', (10,330))
     wx.StaticText(self, -1, 'Var threshold', (10,360))
     wx.StaticText(self, -1, 'Shadow detect.', (10,390))
     wx.StaticText(self, -1, 'num. blobs', (10,410))
     wx.StaticText(self, -1, 'low-pass filter', (10,440))
     wx.StaticText(self, -1, 'delta-X/frame', (10,470))
     wx.StaticText(self, -1, 'pixels for blob', (10,500))


    def on_changed(self, evt):
        #use global variables
     global maxjump
     global history
     global ratiomin
     global ratiomax
     global varThreshold
     global bShadowDetection
     global maxpoints  # maximum number of blobs to track at
once
     global vfilt   # stepwise velocity filter factor (low-
pass filter)
     global maxvel  # maximum physically likely velocity
(delta-X per frame)
     global xdistThresh
     maxjump=self.maxj.GetValue()
```

```
      ratiomin=self.ratmi.GetValue()
      ratiomax=self.ratma.GetValue()
      params.minThreshold=self.minth.GetValue()
      params.maxThreshold=self.maxth.GetValue()
      params.minDistBetweenBlobs=self.distm.GetValue()
      params.filterByArea=self.areaf.GetValue()
      params.minArea=self.areami.GetValue()
      params.maxArea=self.areama.GetValue()
      params.filterByCircularity=self.circu.GetValue()
      params.minCircularity=self.circum.GetValue()
      params.filterByConvexity=self.conv.GetValue()
      params.minConvexity=self.convm.GetValue()
      params.filterByInertia=self.inertia.GetValue()
      params.minInertiaRatio=self.inertiam.GetValue()
      history=self.histo.GetValue()
      varThreshold=self.varthre.GetValue()
      bShadowDetection=self.shadowd.GetValue()
      maxpoints=self.maxpoint.GetValue()
      vfilt=self.vfilt.GetValue()
      maxvel=self.maxvel.GetValue()
      xdistThresh=self.xdistthre.GetValue()

      #print (str(maxjump)+" "+str(ratiomin)+"
"+str(ratiomax)+" "+str(params.minThreshold)+"
"+str(params.maxThreshold)+"
"+str(params.minDistBetweenBlobs)+"
"+str(params.filterByArea)+" "+str(params.minArea)+"
"+str(params.maxArea)+" "+str(params.filterByCircularity)+"
"+str(params.minCircularity)+"
"+str(params.filterByConvexity)+" "+str(params.minConvexity)+"
"+str(params.filterByInertia)+"
"+str(params.minInertiaRatio)+" "+str(history)+"
"+str(varThreshold)+" "+str(bShadowDetection))

    # define the event when click on menu items


    # event quit
    def on_clic_esci(self, evt): quit()

    #save the setting for the next time open
    def on_clic_setting(self,evt):
      global directory
      print "scrivo i valori"
      os.chdir=directory[0]
      setting_file=open("setup.dat", "w")
      self.on_changed
      print (str(maxjump)+" "+str(ratiomin)+"
"+str(ratiomax)+" "+str(params.minThreshold)+"
```

```
"+str(params.maxThreshold)+"
"+str(params.minDistBetweenBlobs)+"
"+str(params.filterByArea)+" "+str(params.minArea)+"
"+str(params.maxArea)+" "+str(params.filterByCircularity)+"
"+str(params.minCircularity)+"
"+str(params.filterByConvexity)+" "+str(params.minConvexity)+"
"+str(params.filterByInertia)+"
"+str(params.minInertiaRatio)+" "+str(history)+"
"+str(varThreshold)+" "+str(bShadowDetection)+"
"+str(self.maxpoint.GetValue())+"
"+str(self.vfilt.GetValue())+" "+str(self.maxvel.GetValue())+"
"+str(self.xdiststhre.GetValue()))
      setting_file.write(str(self.maxj.GetValue())+"
"+str(self.ratmi.GetValue())+" "+str(self.ratma.GetValue())+"
"+str(params.minThreshold)+" "+str(params.maxThreshold)+"
"+str(params.minDistBetweenBlobs)+"
"+str(params.filterByArea)+" "+str(params.minArea)+"
"+str(params.maxArea)+" "+str(params.filterByCircularity)+"
"+str(params.minCircularity)+"
"+str(params.filterByConvexity)+" "+str(params.minConvexity)+"
"+str(params.filterByInertia)+"
"+str(params.minInertiaRatio)+" "+str(self.histo.GetValue())+"
"+str(self.varthre.GetValue())+"
"+str(self.shadowd.GetValue())+"
"+str(self.maxpoint.GetValue())+"
"+str(self.vfilt.GetValue())+" "+str(self.maxvel.GetValue())+"
"+str(self.xdiststhre.GetValue()))
      setting_file.close()

    #event open
    def on_clic_apri(self, evt):
      global directory
      if directory[1]!="":
        os.chdir=directory[1]
      else:
        os.chdir=directory[0]

      #print os.chdir

      openFileDialog = wx.FileDialog(self, "Open", "", "",
                                     "Video files
(*.h264;*.mp4;*.mkv)|*.h264;*.mp4;*.mkv",
                                     #wx.FD_OPEN |
wx.FD_FILE_MUST_EXIST)
                                     wx.FD_OPEN |
wx.FD_MULTIPLE | wx.FD_FILE_MUST_EXIST)
      openFileDialog.SetDirectory(directory[1])
      openFileDialog.ShowModal()
      history=self.histo.GetValue()
```

```
        varThreshold=self.varthre.GetValue()
        bShadowDetection=self.shadowd.GetValue()
        thename = openFileDialog.GetPaths()
        print (len(thename))
        moviename = openFileDialog.GetFilenames()
        print (moviename)
        if thename!="":
            for m in range(len(thename)):
              print ("running tracking ",thename[m])
              tracking(thename[m],moviename[m])

#tracking(thename,moviename)#,history,varThreshold,bShadowDete
ction)
              directory[1]=thename[m].replace(moviename[m],"")
        openFileDialog.Destroy()
        #print os.chdir
        #print directory[1]



#Run the program
if __name__ == "__main__":
    app = wx.App(False)
    frame = MyForm()
    frame.Show()
    app.MainLoop()
```

light.dat file example

This file is read by the python script on Raspberry Pi 3. Each row is a round. The first three numbers are the value for the colors of NeoPixel, respectively Red, Greeen and Blue. The fourth number is the LED number, if it used more than one LED in the same experiment. The last two are the time in second that the Raspberry have to record with the camera, the dark and the light exposure.

```
20 20 20 0 180000 180000
1 1 1 0 180000 180000
15 15 15 0 180000 180000
10 10 10 0 180000 180000
5 5 5 0 180000 180000
```