

Strain-Dependent Optical Transmission in Knitted Fabrics

First Steps in Experimental Characterization and Modeling for
Dynamic Architectural Shading

Research Internship Report

July - August 2026

Elias Tournier Corrigan

Bachelor's student in Microengineering at EPFL

Supervisors and lab:

Juliette Hars and Dr. Sergio Picella

Computational Robot Design & Fabrication Lab (CREATE Lab)



Abstract

This research aims to characterize how light passes through knitted fabrics to project dynamic patterns onto surfaces, inspired by the adaptive skin of cephalopods. By studying how fabric stretching alters light transmission, this project seeks to lay the groundwork for smart, flexible shading materials. Such responsive materials could replace energy-heavy mechanical blinds or help recreate precise lighting conditions for agricultural permaculture and environmental restoration.

To collect precise experimental data, I designed a specialized test setup using motorized mechanisms to stretch knitted fabrics under a controlled light source representing the sun. Using cameras and optical sensors, I measured changes in light intensity and analyzed the resulting visual motifs.

The results demonstrated that the direction in which the fabric is pulled has the greatest influence on the overall light passing through. Additionally, the analysis revealed that certain lighting conditions, such as large areas of dim light, cannot be produced using simple stretching alone.

The next steps for this project involve building a simulation model and expanding testing to various knitting patterns. Ultimately, future work aims to build a two-way mapping between fabric properties and shading outcomes.

Contents

1	Introduction	3
2	Research design	5
2.1	Building an experimental setup	5
2.2	Operating the rig and recording data	8
2.3	Experimental protocol	12
2.4	Developing computational tools and pipelines	12
2.4.1	Unifying the data from multiple runs of the same experiment	13
2.4.2	General lighting analysis	13
2.4.3	Light blobs: detection and tracking	13
2.4.4	Light blobs: individual measurements	14
2.5	Note on the project's timeline and development	14
3	Analysis and findings	14
3.1	Importance of the pulling direction	14
3.2	Light intensity against size of light blobs	15
3.3	Attempt at characterizing the arrangement of the blobs: Ripley's K and L functions	17
4	Conclusion	18
4.1	Next steps	19
4.1.1	Improve experimental setup and desktop app	19
4.1.2	Greater collection of samples and conditions to generalize theories over a vaster dataset	19
4.1.3	Emulating the experiment in a physical simulation	20
4.1.4	Other actuation methods	20
4.1.5	Exploring applications	20
5	Acknowledgements	21

1 Introduction

Before Laidlaw and this summer internship, the only knitwear in my life used to be the Christmas sweaters my grandmother made me. Somehow, microengineering at EPFL brought me back to them, except now I'm interested in what happens when you stretch one and shine a light through it. My attention has shifted from trying to bear their presence on my shoulders to finding sustainable applications of their properties.

Today, shading solutions are often static (fixed blinds, awnings) or expensive and complex (motorized/electronic shading). The UN has adopted the development of "Sustainable cities and communities" (SDG n°11) [1] as one of its 17 Sustainable Development Goals. This goal implies that new solutions should be explored to make our cities and communities more resilient against heatwaves [2] and make better use of natural sunlight, while reducing the energy consumption of our existing infrastructures [3][4]. Shading is therefore an important aspect of this evolution, and research should be conducted on it to find better and more optimal solutions.



Another reason to justify doing research on shading solutions is the development of new agricultural techniques such as permaculture[5], a land-management approach that mimics natural ecosystems. However, before the land has grown into a natural ecosystem, we need to simulate the shading conditions of those environments to help restore the ecosystem. Depending on the species (height, location, etc.), plants are more or less subject to the light of the sun, and new shading solutions could help recreate this variety.

My research focused on knitted fabrics as a possible solution to these problems and requirements. This idea is actually bio-inspired: like cephalopod skin that uses muscle-actuated papillae to change light scattering[6], knitted

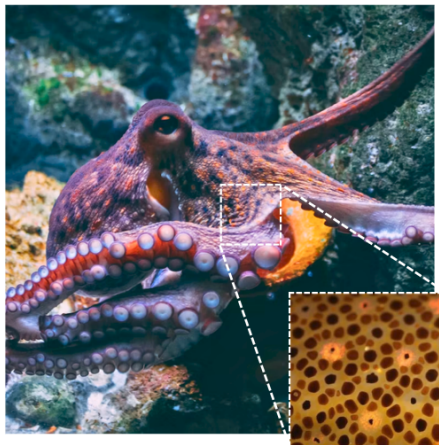


Figure 1: Picture of a cephalopod and a zoom on cephalopod skin

structures deform under tensile strain (ϵ), changing loop aperture geometry and acting as a tunable lens array that shapes light into blobs.

Ultimately, this research subject tries to answer the question: can we predict, and eventually control, the shading pattern a knitted fabric produces, given the physical conditions of the fabric and the characteristics of the light source? This question introduces the concepts of the forward model, which would predict the resulting lighting pattern depending on the knit and conditions, and the inverse model, which would determine which patterns, under which conditions, work to get a required shading pattern.

$$S = f(K, C)$$

$$\{(K, C)\} = f^{-1}(S_{target})$$

Where S are the properties of the projected shade, K the parameters of the knit, and C the luminosity (light angle, ...) and mechanical (strain in the fabric) conditions.

For the first steps in our research, we primarily focused on how stretching affects the light transmitted through a knitted fabric under direct light. Our goal was mainly to create the test setup and do the first analyses to check its reliability. Therefore, we didn't have a wide collection of different samples and mainly studied a 13 cm x 13 cm fabric sample that used the stockinette stitch.

To designate the spatial orientation of the knits, we used the knitting vocabulary. The directions in which the sample is stretched can be described by "courses", which represents the direction in which the fabric runs, or "wales", which is the direction perpendicular to it[7]. Our sample was knitted using the

“stockinette” stitch. With this knitting pattern, the side facing the light can be described using “knit”, which designates the smooth and normally exposed side of the knit (for clothing), or “purl”, which is the bumpy side of the stitches on the other side.

We studied the influence of strain on the transmitted light. Strain describes to what extent a material is stretched compared to its size when it’s not under any load.

$$\epsilon = \frac{L - L_0}{L_0}$$

Where L is the length of the material under load and L_0 is the length of the sample at rest.

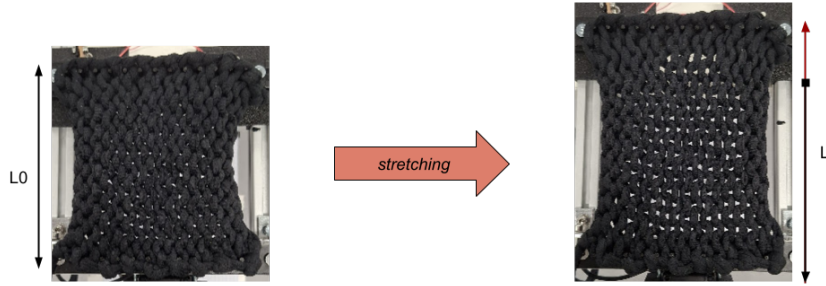


Figure 2: Impact of stretching the fabric on the loop-aperture of the knit

2 Research design

2.1 Building an experimental setup

When we started the project, we didn’t have an existing rig to both stretch the fabric and do optical measurements. Therefore, the first part of the internship

was building this experimental setup. We wanted it to be operated digitally so that the measurement and the actuation could be coupled in one software, and one computer could be used to both record the data and operate the rig.

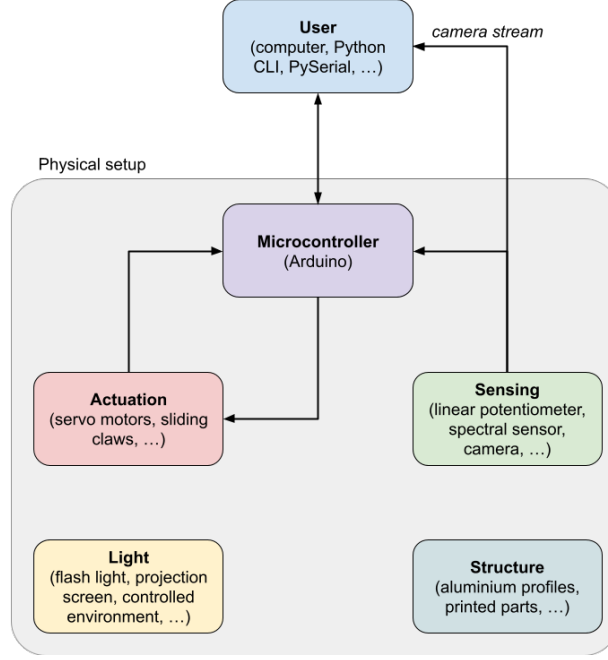


Figure 3: Diagram of the experimental setup modules and system dependencies

We started by building the mechanism to stretch the fabric. Using aluminium profiles and printed parts, we created a slider and claws on which the sample can be hooked. Motors behind the sliding claw pull a string attached to it and stretch the fabric. We put a stand so that we could attach a strong flashlight above and shine light directly on it. The flashlight has an optical configuration that mimics the sun’s distance to Earth, as the incident rays are nearly parallel to each other.

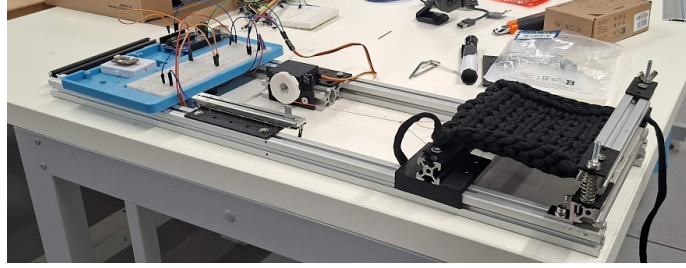


Figure 4: Photo of the rig during its early construction stage: microcontroller (on the left), motor (center), claws and sample (right)

We placed a projection screen underneath to be able to visualize the transmitted light. We placed the light sensor (Adafruit 10-channel spectral sensor) and cameras underneath. This helped to avoid saturation of the optical sensors, capture the shading patterns across a plane area, and maintain uniform camera distance. We then built a box to contain the experiment so that no light could shine through and make the results dependent on the lab's lighting conditions.

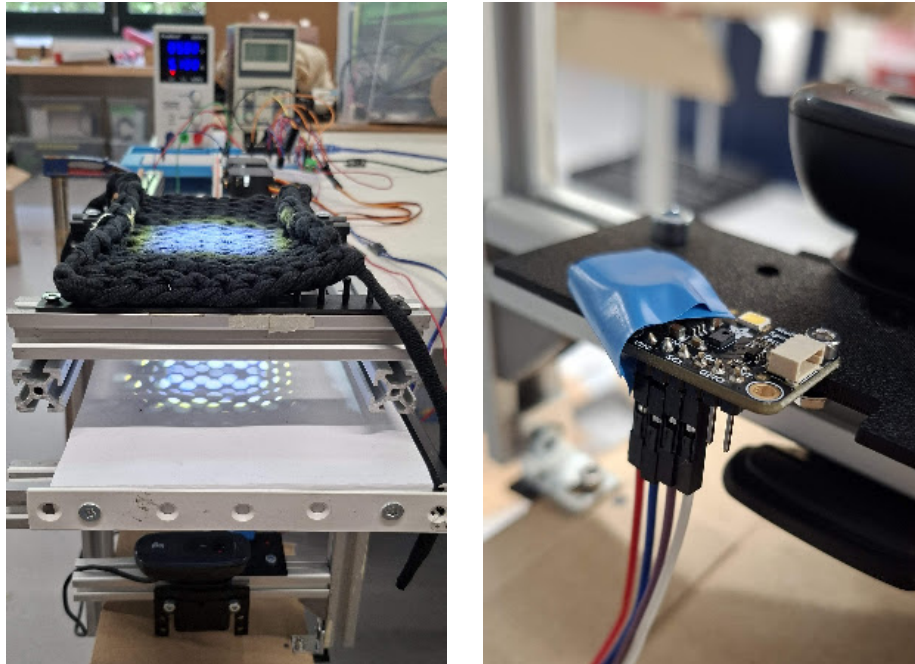


Figure 5: Pictures of the setup taken from the front with the camera under the screen (left) and of the Adafruit spectral sensor (right)

The advancement of the sliding claw is measured using a linear potentiometer that has been calibrated to get linear measurements and to know exactly the spatial position of the slider.

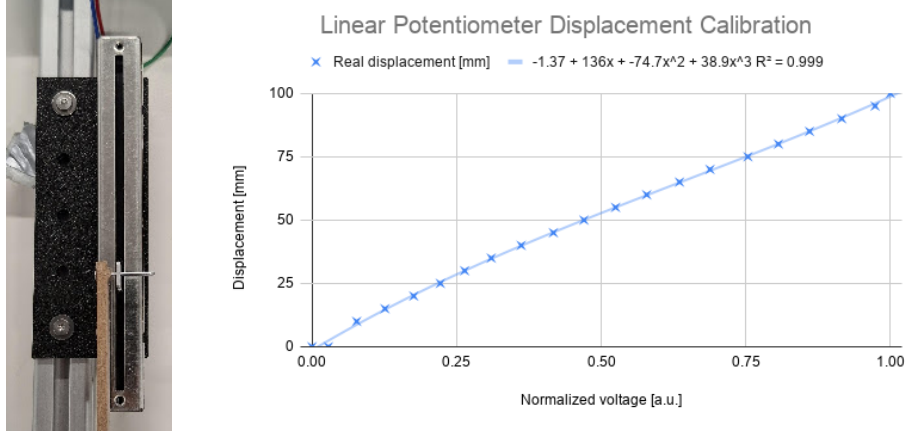


Figure 6: Linear potentiometer and its calibration plot to measure the applied strain

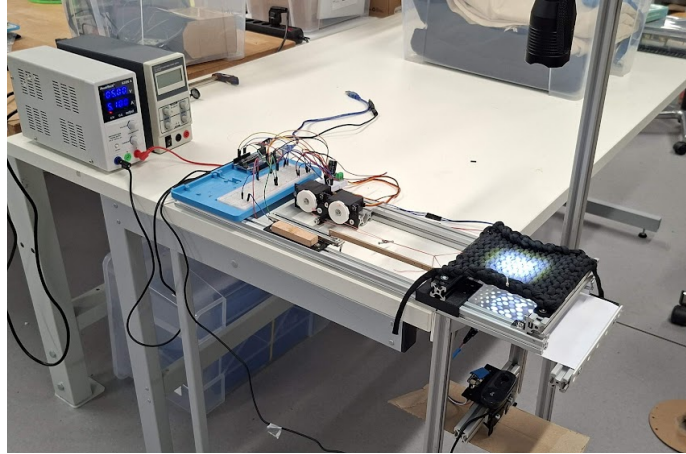


Figure 7: Picture of the experimental setup at the end of the project (without the enclosing box)

2.2 Operating the rig and recording data

The rig operating system is divided into two parts: a microcontroller on the rig, which is responsible for controlling the motors and reading the raw values of

the sensors, and a desktop application that enables the user to send commands to the rig and record the data from the sensors. Both communicate through a USB cable connecting the microcontroller to the computer using serial communication.

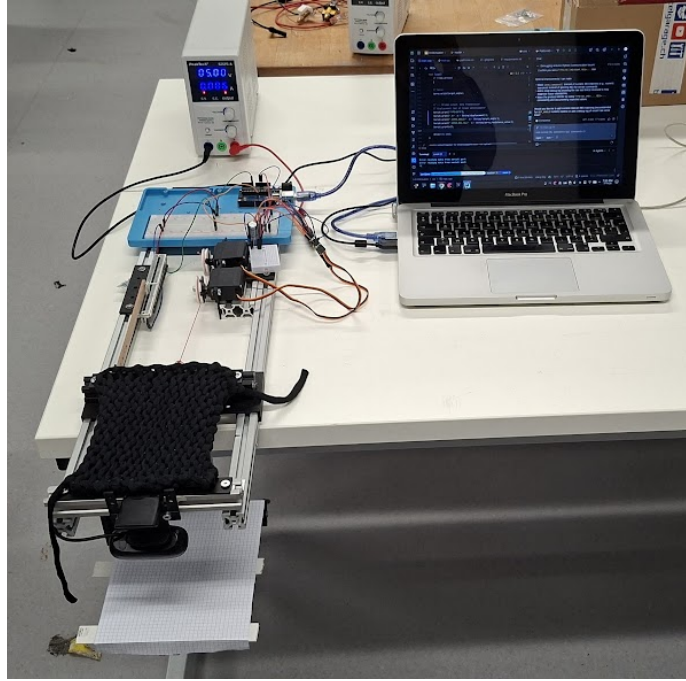


Figure 8: Experiment (left) and computer running the tools (right) during early stages of development

The rig is operated using an Arduino microcontroller. The different modules of the rig are wired through soldered connections and a breadboard. The Arduino is powered thanks to the USB connection, while the motors are powered using a lab power source. During the development of the rig, we had issues with powering the motors and had to add capacitors to the circuitry to stabilize the current. The raw values of the sensors are read by the microcontroller, but the camera image feed is directly read by the computer.

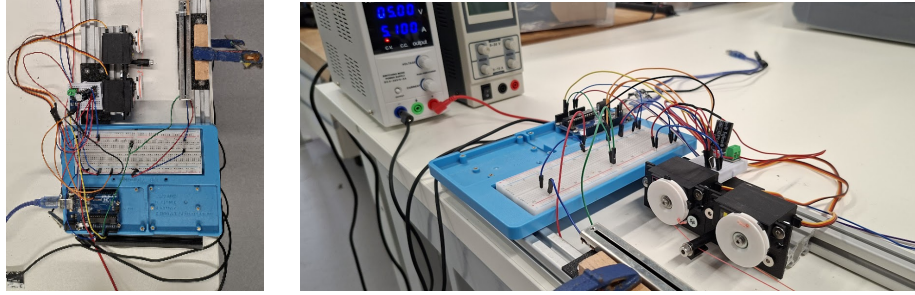


Figure 9: Pictures of the electronic and electrical circuitry: the microcontroller, the breadboard, the motors, and the lab power supply

The desktop application was developed in Python, a very popular programming language, which will make continuing the project for other students easier. The PySerial library was used. It works like a code “extension” that enabled us to interact with the microcontroller: we can read the messages from the microcontroller and send messages to it. For the first steps, the desktop app works as a “Command Line Interface” (CLI) tool, meaning users have to type in commands to execute tasks.

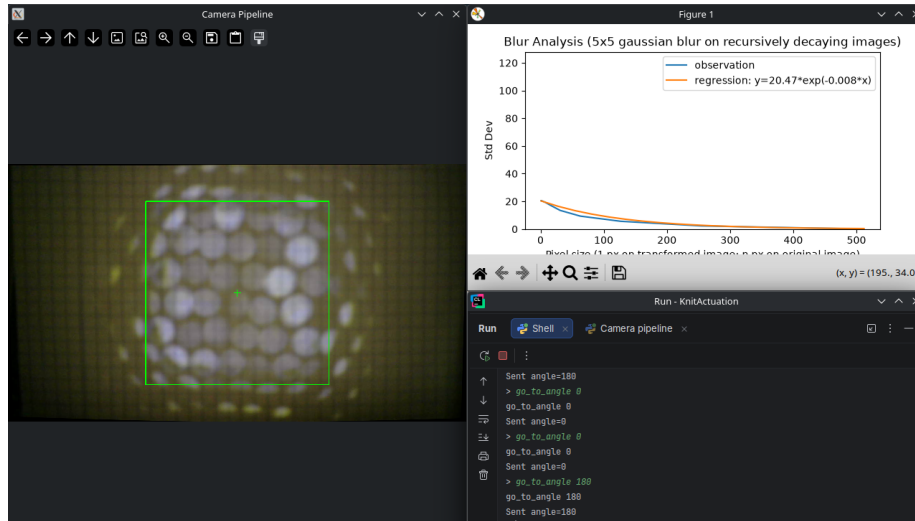


Figure 10: Screenshot of the desktop app interface with the camera stream’s preview, the CLI, and a test analysis of the picture

Making the communication between the computer and the Arduino robust and efficient wasn’t a simple task. A full protocol was developed using a

two-phase confirmation system. The full workflow works as follows. The human user sends a command in the tool, and the tool then interprets it. Depending on the command, it sends the appropriate task execution request to the microcontroller. When the microcontroller receives the message, it checks if it understands and knows what it has to do as a consequence. If it understands, it responds with an acknowledgement message stating what it understood, and if not, it returns an error detailing the command with which it had an issue. Lastly, the desktop app checks if what the microcontroller understood was indeed what was expected of it. Overall, developing this protocol helped make the rig more reliable and avoid errors in data logging.

The camera was operated using OpenCV[8], a Python library that enables us to read the images recorded by the camera, change its optical and technical settings, and later do some analysis on the images.

The developed application made recording data easier. Instead of having to manually enter the data in spreadsheet tables, the software uses a system of “experimental runs”, automatically saving experimental results in folders and also saving the experiment’s metadata (name, angle of light, date, etc.). This made the transition from recording to analyzing the data a seamless process.

For later studies, we also wanted to record the force needed to stretch the fabric using a “shunt” current sensor; however, the values were incoherent and unexploitable. This was certainly due to the use of a breadboard for this sensor, which didn’t guarantee electrical connections of good quality.

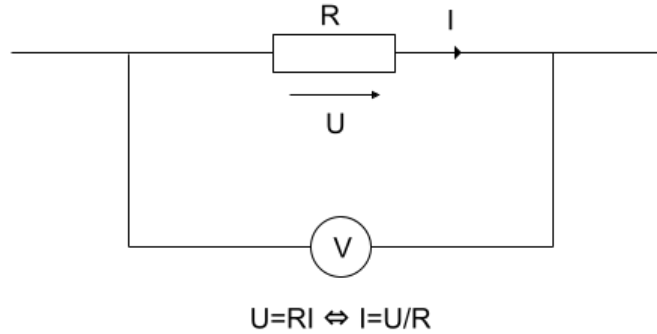


Figure 11: Shunt current sensor electrical principle

2.3 Experimental protocol

An important part of any experimental research project is developing an experimental protocol and documenting it so that other researchers who will continue or build on your research know how you got your data.

When working with a physical experimental setup that deals with mechanical pieces, there is a tendency for experimental runs not to be independent because of the hysteresis of mechanical properties. Therefore, to make each run independent, the setup needs to be reset, and the run needs to be done under similar conditions.

We conducted each experiment several times over several runs and also measured significant states such as the blank state (no sample, direct light) to check the significance of our values (check if mechanical components of the experimental setup were influencing the result and nullifying their effect in other experiments if they were).

The experiments we conducted were mainly done on one sample, as the main goal wasn't to conclude generalities from our analysis but to check the reliability of the rig.

The analyses we conducted were then done on averages over the several runs, and metrics were also computed to check the coherence of the data and thus check if the experiments were reproducible.

2.4 Developing computational tools and pipelines

The next major step in the research project was developing computational tools and a series of pipelines to measure and characterize the data we were going to record. Again, the Python programming language was used for readability and the number of already existing tools and the scale of the documentation for data science in this language. We used Jupyter Notebooks to run and document the analysis code. The system of running independent cells made analysis efficient, as we didn't have to rerun the whole code again to change one simple value or the rendering of a plot. The presence of Markdown, a markup language, made the documentation and segmentation of the code easier, making it easier to navigate the different parts. A few different pipelines were developed to clean the data and experiment with some initial analyses in the global project. Indeed, one of the main objectives of conducting analysis on the data during the early steps of a project is to check the reliability of the experimental setup. The pipeline we developed would help to analyse trends and confirm the hypotheses we had formulated when observing the behavior of the transmitted light when we stretched the fabric.

2.4.1 Unifying the data from multiple runs of the same experiment

The goal of this pipeline is to cluster the different data points of the different runs of one experiment according to strain values that are close to each other. Then, with those clusters, the pipeline computes the averages and standard deviations of the clustered data points to create one dataset for the whole experiment. The standard deviation helps to assess the reproducibility of the runs and the quality of the data; thus, it also helps us evaluate the quality of the setup to record experimental data.

2.4.2 General lighting analysis

Using the computed data in the unifying data step and using tools such as NumPy[9] for data analysis and OpenCV for image processing, we built a simple general pipeline to measure the intensity of the transmitted light depending on the strain level. This pipeline uses the data from the Adafruit spectral sensor and the camera. The spectral sensor returns the intensity of the light at different wavelengths, but using the “clear” channel gives an idea of the ambient luminosity, a mix of all channels. Then, converting the recorded color pictures to grayscale images and calculating the average gray value of the picture also gives us a way of measuring the intensity of the transmitted light. Limitations of using the data from these sensors are going to be discussed in the conclusion. Then, the different formulas are used on the data to compute the desired measurements (absolute, relative, normalized, etc.).

2.4.3 Light blobs: detection and tracking

Using the averaged pictures and image processing tools from SciKit-Image[10], this pipeline analyzes the images to detect the blobs of light projected by the light passing through the fabric. After converting the image to grayscale, normalizing it, and slightly blurring it to simplify processing, the detection of blobs was made easy thanks to the “Laplacian of Gaussian” algorithm[11][12]. The pipeline first outputs the coordinates of the positions and the sizes of the blobs in the images.

Because the setup pulls asymmetrically on the fabric (only from one side), the blobs in the pictures move in one direction the more you stretch the fabric. To compensate for that, the pipeline tracks the blobs in the image by comparing the positions of the blobs between nearby strain intensity levels. In the process, the algorithm measures the average displacement and clips the image in the opposite direction to keep the blobs standing still as you stretch the fabric.

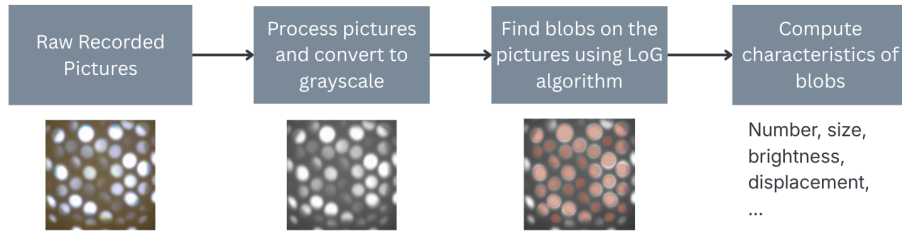


Figure 12: Diagram of the pipeline with the evolution of the pictures at each step

2.4.4 Light blobs: individual measurements

Thanks to the outputs of the light blob detection, we can further process the images to extract other information such as the light intensity of individual blobs. By knowing the position of each blob and its radius, we can compute the average intensity of the pixels that represent the blobs on the image.

2.5 Note on the project's timeline and development

Although these steps are presented in chronological order, the development of the different modules was a continuous process, and progress in certain areas revealed weaknesses in others. For example, when we started analyzing the data, we discovered it was maybe better to improve our protocol and make some modifications to the test setup to have data of better quality. Not all analyses were chosen from the start of the project, as new hypotheses were formulated as we progressed and made new observations.

3 Analysis and findings

During the project, we tried many different analyses with the different tools we had developed to try to identify properties and variables that evolved significantly as a function of the strain. The small scale of the recorded dataset and sample collection made it difficult to generalise our analyses, but we were still able to confirm some of our hypotheses. The analyses that were developed can be separated into two categories: global/broad and local/detailed. Here are a few analyses we have conducted and the results we obtained. The results were rendered using Matplotlib[13], a Python data visualisation library.

3.1 Importance of the pulling direction

We have conducted a total of four different experiments on the same sample (knitted using the Stockinette stitch): 2 different pulling directions and 2 different sides that could be combined. Our initial hypothesis was that the pulling

direction was the main factor influencing how the transmitted light behaved depending on the strain and that which side faced the light wasn't a significant factor. To show that, we plotted the increase in the transmitted light intensity of the spectral sensors and the images against the increase in strain in the fabric. On the plots we obtained, we can see that the two experiments with the same pulling direction (courses or wales) share a very similar trend in the evolution of the light intensity. However, a significant trend cannot be observed for the two experiments with the same side facing the light.

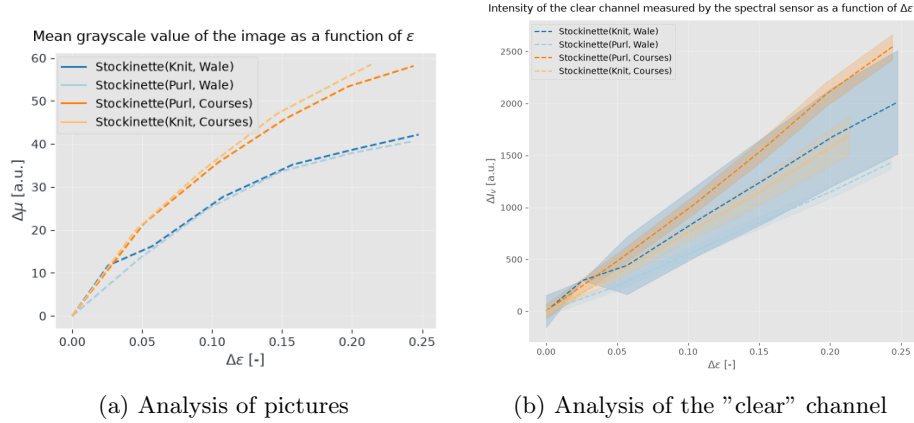


Figure 13: Two analyses of the evolution of brightness as a function of $\Delta\epsilon$ that use different data from different sensors: camera (a) and spectral sensor (b)

Thus, we can conclude that, with this sample, it is indeed the pulling direction that is the significant factor of change in luminosity.

3.2 Light intensity against size of light blobs

One of the applications of this project would be to create programmable shading. The light blobs we observed on the screen could be potential candidates to develop this, as their intensity and size can seem to evolve logically depending on the applied strain. Thus, one of the analyses we wanted to do to characterize the blobs was to see how their light intensity evolves with their size. Using the pipelines for the detection and measurement of light blobs, we plotted for each experiment the brightness of each blob against its size.

Characterisation of individual blobs on the pictures as a function of σ and ε for 'Stockinette(Purl, Wale)'

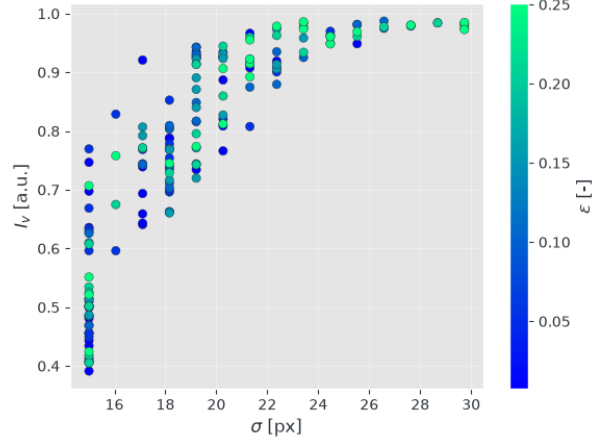


Figure 14: Sample analysis of individual blobs

The whole plot is mostly confined to a diagonal band: very bright small blobs and very dim large blobs are both rare, meaning brightness and size are tightly coupled across the whole range. This seems logical, as the strings that compose the knit are opaque, and thus when an aperture in the fabric is bigger, it creates both a bigger blob and lets more light through.

In addition to analysing the blobs individually, we can also analyze the average blobs at each actuation level.

Characterisation of average blobs on the pictures as a function of σ and ϵ for 'Stockinette(Purl, Wale)'

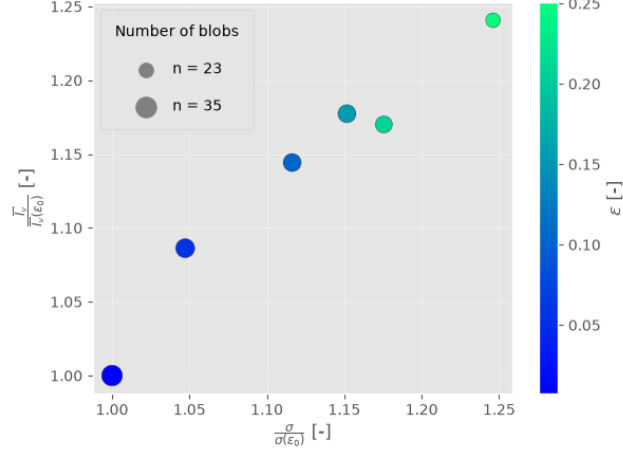


Figure 15: Sample analysis of average blobs with normalized data

From this plot, we can see that the average blob's size and brightness are linearly coupled, materialised by the alignment of the data points.

3.3 Attempt at characterizing the arrangement of the blobs: Ripley's K and L functions

Lastly, we tried characterizing the arrangement and structure of the blobs using their positions. For that, we first wanted to show that the structure wasn't random using Ripley's K and L spatial analysis[14]. Ripley's K and L functions calculate the spatial clustering, dispersion, or randomness of point patterns across multiple distance scales.

$$K(r) = \lambda^{-1} \sum_{i \neq j} \frac{I(d_{ij} < r)}{n}$$

with λ that represents the average concentration of light blobs in the picture, d_{ij} the distance between blobs i and j , I an indicator function that returns 1 if $d_{ij} < r$ or else 0, and n the number of detected blobs.

$$L(r) = \sqrt{\frac{K(r)}{\pi}}$$

Isotropic Ripley's L analysis of the images of the "Stockinette(Purl, Courses)" experiment for different actuation levels

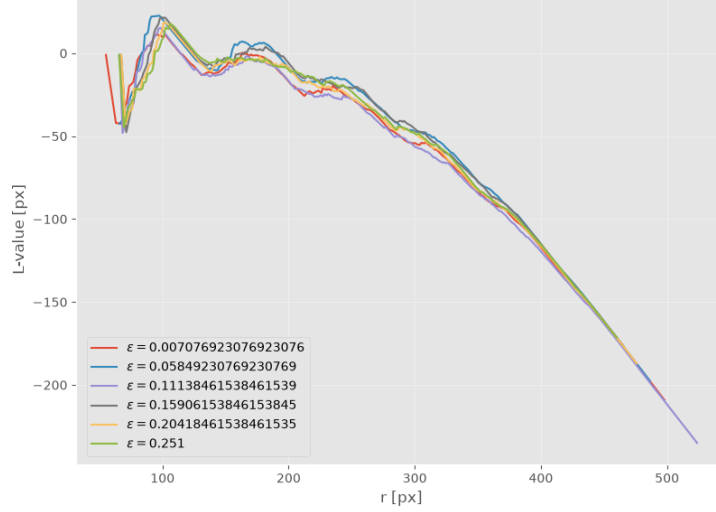


Figure 16: Note: the plotted quantity is $y = L(r) - r$, a detrended form that makes deviation from spatial randomness easier to read directly as a deviation from zero.

If the curves had followed a straight line, then the structure would be random; however, since it is a wavy curve at the beginning, this means that the blobs have a structure.

However, for the moment, this analysis isn't valid because it is naturally biased by the blobs closer to the edges of the picture that have fewer blobs around them. Because points near the boundary of the image have unobserved neighbors outside the study region, an edge-correction weight should be added to compensate for that[15].

4 Conclusion

During this research project, we followed the first steps of a longer journey to determine if it is possible to create dynamic and programmable shading solutions using knitted fabrics. These first steps consisted of building a physical experimental setup, creating software tools to record data, and developing different analyses of the collected optical data. For my first research experience, it was also about learning processes: development processes, experimental protocols, and imagining future steps. The experimental setup has enabled us to efficiently record data following the protocol we have established, and the analyses have shown that it is reliable and that the experiments are reproducible. Moreover, the analyses have produced some interesting results that we hope to generalize

later. This is an already well-advanced starting point, but it is clear we are still at the genesis of the project, and many experiments are still to be tried; the experimental environment and analyses need to be improved, and many other hypotheses are to be verified and generalized. The final goal of building mathematical models is still far, but we already have a base to build upon.

4.1 Next steps

4.1.1 Improve experimental setup and desktop app

Many things can still be improved in the experimental setup, such as adding more configuration options to the flashlight (angle, intensity, ...), but also being able to stretch the fabric in more complex ways (ex., from two different directions at the same time). Moreover, most of the data we collected isn't expressed in conventional units but in arbitrary units instead because the sensors we used don't return absolute values. Instead of using the clear channel of the Adafruit spectral sensor, it could be interesting to use a luxmeter and compare the transmitted light to the theoretical brightness of the lamp.

The app could also be improved to make data collection and the operation of the rig easier. Indeed, features to automate the recording of different runs could be added to facilitate the experimental process. In addition, the app could benefit from having a real interface instead of the user having to enter commands in a command-line interface.

4.1.2 Greater collection of samples and conditions to generalize theories over a vaster dataset

For the moment, all data recording and analyses have been made for multiple experiments performed on one sample. Therefore, we cannot advance any theory or model because our observations are too limited. Many parameters can be changed to create a vast collection of samples. For instance, the stitch pattern and yarn properties, and their impact on the transmitted light, could also be interesting variables to study.

In our study, we have analyzed the effect of direct light and shown that, on our sample, the pulling direction was the more significant factor. However, the sun doesn't always shine directly on our blinds, and it will be interesting to see how the 3D structure of the knit impacts the transmitted light when it arrives incidently at an angle on the fabric.

Beyond being able to generalize theories and hypotheses, having a bigger dataset could help train AI models and use powerful data science tools that rely on big datasets.

4.1.3 Emulating the experiment in a physical simulation

The need for creating a big dataset and testing many different samples also poses issues. Indeed, physical samples are not easy to make; they take time to knit and are imperfect. Thus, it has become quite clear towards the end of the project that if we wanted any significant progress to be made on the different questions linked to this subject, we would have to simulate the experiments we're currently conducting in the real world on a computer. Beyond being able to acquire more data, using simulations could help us use computational methods to create models, such as optimisation algorithms.

4.1.4 Other actuation methods

Our experimental setup uses motors to stretch the fabric; however, it could be interesting to explore other actuation methods. For example, research in new domains such as soft robotics has theorized the use of soft motors that can be pneumatically actuated. These are called pouch motors[16].

4.1.5 Exploring applications

This subject of research is quite new and is part of a broader domain that is emerging in robotics called soft robotics. We have identified some potential applications of our topic, but we believe many more can be found. Changing the scale to smaller dimensions could help create dynamic filters for domains such as fluid mechanics.

5 Acknowledgements

I would like to express my deepest gratitude to my internship supervisors, Juliette Hars and Dr. Sergio Picella, for their invaluable patience and the feedback they have given me during the project. I am also extremely grateful to Prof. Josie Hughes for accepting me into her lab and enabling me to work on one of her PhD students' projects. I would like to extend my sincere thanks to my colleagues, with whom working in the lab was very enjoyable and an enriching experience.

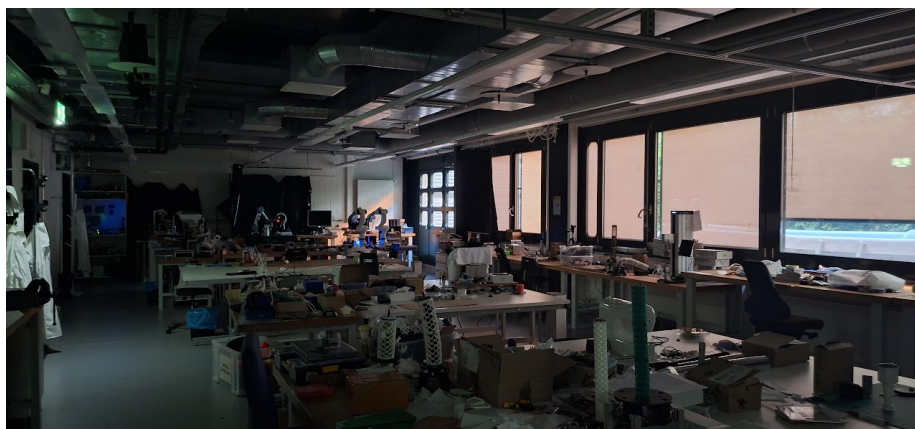


Figure 17: Leaving the lab late in the evening

References

- [1] United Nations General Assembly. *Transforming Our World: The 2030 Agenda for Sustainable Development (Goal 11: Sustainable Cities and Communities)*. Accessed: October 1, 2026. Oct. 2015.
- [2] Katherine Calvin et al. *IPCC, 2023: Climate Change 2023: Synthesis Report. Contribution of Working Groups I, II and III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change [Core Writing Team, H. Lee and J. Romero (eds.)]*. IPCC, Geneva, Switzerland. Tech. rep. Intergovernmental Panel on Climate Change (IPCC), July 2023.
- [3] Negin Imani et al. “Biomimetic Shading Systems: Integrating Motorised and Moisture-Responsive Actuation for Adaptive Façades”. In: *Biomimetics* 10.10 (Oct. 2025), p. 711.
- [4] Pelin Avcı, Berk Ekici, and Zehra Tuğçe Kazanasmaz. “A review on adaptive and non-adaptive shading devices for sustainable buildings”. en. In: *Journal of Building Engineering* 100 (Apr. 2025), p. 111701.
- [5] Bill Mollison and David Holmgren. *Permaculture One*. 1978.
- [6] Justine J. Allen et al. “Comparative morphology of changeable skin papillae in octopus and cuttlefish: Comparative Morphology of Changeable Skin Papillae”. en. In: *Journal of Morphology* 275.4 (Apr. 2014), pp. 371–390.
- [7] *Knitted fabric*. en. Page Version ID: 1362298099. July 2026.
- [8] G. Bradski. “The OpenCV Library”. In: *Dr. Dobb’s Journal of Software Tools* (2000).
- [9] Charles R. Harris et al. “Array programming with NumPy”. en. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362.
- [10] Stéfan Van Der Walt et al. “scikit-image: image processing in Python”. en. In: *PeerJ* 2 (June 2014), e453.
- [11] Tony Lindeberg. “Feature Detection with Automatic Scale Selection”. en. In: *International Journal of Computer Vision* 30.2 (Nov. 1998), pp. 79–116.
- [12] *Blob Detection — skimage 0.26.0 documentation*.
- [13] John D. Hunter. “Matplotlib: A 2D Graphics Environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95.
- [14] B. D. Ripley. “Modelling Spatial Patterns”. en. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 39.2 (Jan. 1977), pp. 172–192.
- [15] François Goreaud and Raphaël Péliissier. “On explicit formulas of edge effect correction for Ripley’s K -function”. en. In: *Journal of Vegetation Science* 10.3 (June 1999), pp. 433–438.
- [16] Ryuma Niiyama, Daniela Rus, and Sangbae Kim. “Pouch Motors: Printable/inflatable soft actuators for robotics”. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. Hong Kong, China: IEEE, May 2014, pp. 6332–6337.