



ELSEVIER

Contents lists available at [ScienceDirect](https://www.sciencedirect.com)

HardwareX

journal homepage: www.elsevier.com/locate/ohx

A plug-and-play Hyperspectral Imaging Sensor using low-cost equipment



Jairo Salazar-Vazquez, Andres Mendez-Vazquez

Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Av. del Bosque 1145, colonia el Bajío, ZC, 45019, Mexico

ARTICLE INFO

Keywords:

Low-cost HSI
Imaging Spectroscopy
Hyperspectral Imaging Sensor
HSI
Hyperspectral application

ABSTRACT

Hyperspectral Imaging Sensors (HSI) obtain spectral information from an object, and they are used to solve problems in Remote Sensing, Food Analysis, Precision Agriculture, and others. This paper took advantage of modern high-resolution cameras, electronics, and optics to develop a robust, low-cost, and easy to assemble HSI device. This device could be used to evaluate new algorithms for hyperspectral image analysis and explore its feasibility to develop new applications on a low-budget. It weighs up to 300 g, detects wavelengths from 400 nm–1052 nm, and generates up to 315 different wavebands with a spectral resolution up to 2.0698 nm. Its spatial resolution of 116×110 pixels works for many applications. Furthermore, with only 2% of the cost of commercial HSI devices with similar characteristics, it has shown high spectral accuracy in controlled light conditions as well as ambient light conditions. Unlike related works, the proposed HSI system includes a framework to build the proposed HSI from scratch. This framework decreases the complexity of building an HSI device as well as the processing time. It contains every needed 3D model, a calibration method, the image acquisition software, and the methodology to build and calibrate the proposed HSI device. Therefore, the proposed HSI system is portable, reusable, and lightweight.

© 2019 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Specifications table:

Hardware name	Insert hardware name
Subject area	<ul style="list-style-type: none"> • Hyperspectral Imaging • Environmental and Agricultural Sciences • Educational Tools and Open Source Alternatives to Existing Infrastructure
Hardware type	<ul style="list-style-type: none"> • Imaging tools • Measuring physical properties and in-lab sensors • Field measurements and sensors • Electrical engineering and computer science
Open source license	GNU General Public License (GPL)
Cost of hardware	\$550 USD
Source file repository	https://osf.io/s7jzu/

E-mail addresses: jsalazar@gdl.cinvestav.mx (J. Salazar-Vazquez), amendez@gdl.cinvestav.mx (A. Mendez-Vazquez)

URL: <https://gdl.cinvestav.mx/> (J. Salazar-Vazquez), <https://gdl.cinvestav.mx/> (A. Mendez-Vazquez)

URLs: <https://gdl.cinvestav.mx/> (J. Salazar-Vazquez), <https://gdl.cinvestav.mx/> (A. Mendez-Vazquez)

<https://doi.org/10.1016/j.ohx.2019.e00087>

2468-0672/© 2019 The Author(s). Published by Elsevier Ltd.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Hardware in context

The Hyperspectral Imaging Sensors (HSI) allows to acquire Hyperspectral Images (HI). A HI is a collection of digital images acquired simultaneously from a physical surface. Any of those images displays the electromagnetic energy sensed at a specific wavelength. Typically, an HSI senses from 400 nm to 2,500 nm, with continuous and narrow separation of 4–10 nm. Those images are also known as wavebands. A set of wavebands is represented by λ . The Field of View (FOV) of every waveband in λ is aligned spatially. Consequently, a pixel p_i in a HI, has $L = |\lambda|$ components, where $|\cdot|$ is the cardinality of a set. The value of the components in a pixel are related to the materials in its sensed area. Therefore, if same material is sensed in two different pixels (p_i, p_j), then, $\|p_i - p_j\| \simeq 0$, where $\|\cdot\|$ is the *Euclidean Distance*. It enables the Hyperspectral Image Analysis (HIA) to detect, identify, and distinguish materials. This technology has shown its capability to solve problems of Precision Agriculture [1–3], Remote Sensing [4,5], Manufacturing [6], and others.

Although it is possible to acquire commercial HSIs, prices start at 28,000 dollars, making them quite restrictive for many potential users and applications. Thus, the need for developing low-cost HSI systems. Furthermore, with more researchers and engineers wanting to be involved in the development of HIS applications, they are a natural target for this low-cost HI systems. In the literature [7–14], it is possible to find many possible examples of such developments. However, the assembling and use of those prototypes require to have specialized skills on optics, invest a long time in the assembling process, and develop software for calibration and hyperspectral image acquisition.

2. Hardware description

The proposed HSI device can export hyperspectral images with 116×110 pixels, up to 315 wavebands from 400–1052 nm, and a size of 35 Mb. This device has a total weight of 300 g, with a dimension of $18 \times 9 \times 7$ cm (Fig. 1). We have included those characteristics in the comparison between the proposed HSI and the devices at the literature (Table 1). The parameters of the other HSI devices appear in their corresponding reference paper. Our model has shown to be better in terms of cost, weight, and spectral resolution.

Moreover, we analyzed their spectral range, which expresses the shorter and longer wavelength detectable for a specific HSI. We did it because some materials are distinguishable on specific wavelengths only. Therefore, it is desirable to have an HSI with a wide spectral range. For example, the proposed HSI spans the VIS spectrum and part of the NIR spectrum. Although there is an HSI [11] that has a more extensive spectral range, the high cost of the imaging camera makes it out of the scope of low-cost HSIs.

Additionally, mobile optomechanical parts reduce HSI's life-span, as well as it happens in other devices [8–10,12]. In comparison, the proposed HSI device does not possess movable parts, making it possible to increase its life-span and decreases possible errors due to the vibrations. Furthermore, the size and light-weight make the proposed HSI device suitable for many applications as payload in Unmanned Aerial Vehicles (UAV). There is another HSI device [7] with a smaller size, but its size depends on the application at hand.

Unlike the similar works, we have introduced a framework to build the proposed HSI from scratch. This framework decreases the complexity of building an HSI device, as well as the invested time. It includes every needed 3D model, a calibration method, the image acquisition software, and the methodology to build and calibrate the proposed HSI device. To facilitate the use of HSI technology, we have used only low-cost equipment which is available online. For example, it uses a Raspberry Pi 3 Model B+ (RPI) and a Raspicam NoIR v2.1. Furthermore, we deployed the software in an executable file which is compatible with almost any version of Linux (.Appimage [15]). Moreover, the parts of the case were printed by a thermoplastic 3D printer to reduce weight and cost. This technology also reduces the spent time because it is an alternative to sophisticated Computer Numerical Control (CNC) machines. Thus the proposed framework will enable any researcher to assemble his own HSI, investing a short time and a low-budget. Both the proposed HSI and the proposed framework have an open-source hardware and software license, which enables everybody to use, modify, and improve the initial prototype.

Therefore, the researchers could be able to:

- Develop algorithms for hyperspectral image analysis.
- Create the Computer Hyperspectral Stereo Vision.
- Detect and identify different materials.
- Develop hyperspectral 3D reconstruction.
- Be involved in novel UAV applications.

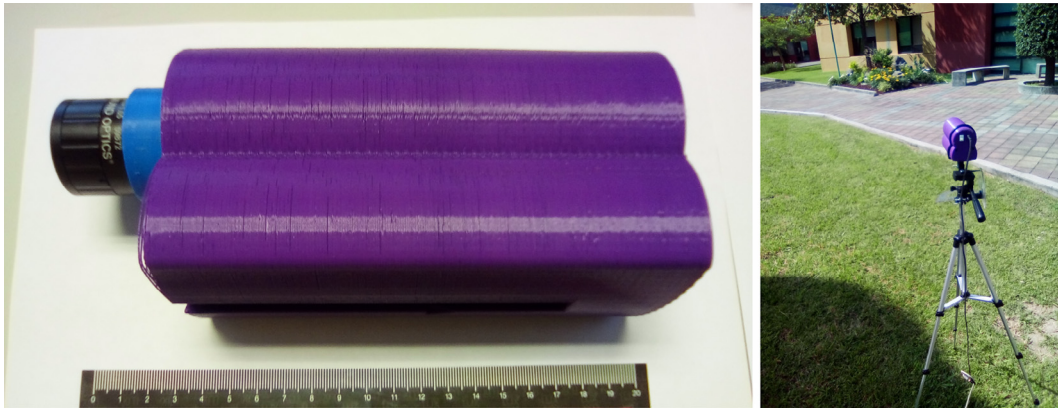


Fig. 1. Assembled prototype.

3. Design files

3.1. Design files summary

Design filename	File type	Open source license	Location of the file
<i>Part_01 Case</i>	3D Model (.stl)	GPL GNU	https://osf.io/etkx3/
<i>Part_02 Front lens</i>	3D Model (.stl)	GPL GNU	https://osf.io/v3e8y/
<i>Part_03 Square aperture</i>	3D Model (.stl)	GPL GNU	https://osf.io/g8n5u/
<i>Part_04 Extension 1 of 2</i>	3D Model (.stl)	GPL GNU	https://osf.io/3jn24/
<i>Part_05 Extension 2 of 2</i>	3D Model (.stl)	GPL GNU	https://osf.io/sk4d8/
<i>Part_06 Diffraction grating</i>	3D Model (.stl)	GPL GNU	https://osf.io/vyxd6/
<i>Part_07 Camera base</i>	3D Model (.stl)	GPL GNU	https://osf.io/3485y/
<i>Part_08 Seal extension</i>	3D Model (.stl)	GPL GNU	https://osf.io/xjy7f/
<i>Part_09 X extension</i>	3D Model (.stl)	GPL GNU	https://osf.io/tkng5/
<i>Part_10 Lid</i>	3D Model (.stl)	GPL GNU	https://osf.io/k67sf/
<i>Part_11 Hanger</i>	3D Model (.stl)	GPL GNU	https://osf.io/wdpc6/
<i>Part_12 Hanger clip</i>	3D Model (.stl)	GPL GNU	https://osf.io/cnrwp/
<i>Part_E02 Editable front lens</i>	CAD (.sldprt)	GPL GNU	https://osf.io/4z9ac/
<i>Part_E08 Editable seal extension</i>	CAD (.sldprt)	GPL GNU	https://osf.io/4z9ac/
<i>Part_E10 Editable lid</i>	CAD (.sldprt)	GPL GNU	https://osf.io/jq67a/

- The *Part_01 Case* protects the electronics and optics in the HSI. Moreover, it has three internal rows to ensure the right assembling.
- The *Part_02 Front lens* is the frontal lenses mount.
- The *Part_03 Square aperture* allows to control the amount of light admitted through the optical system.
- The *Part_04 Extension 1* and *Part_05 Extension 2* are two tubes with a total length of 100 mm. These parts guaranties the required separation between the frontal lens and the diffraction grating.
- The *Part_06 Diffraction grating* is a diffraction grating mount.
- The *Part_07 Camera base* is a Raspicam mount.
- The *Part_08 Seal extension* is a tube that connects the *Part_07 Camera base* and the *Part_10 Lid* to immobilize the parts of the optical system.
- The *Part_09 X extension* immobilizes the RPI.
- The *Part_10 Lid* compresses the *Part_08 Seal extension* and the *Part_09 X extension*. It encloses the case to immobilize the optical system and the electronics.
- The *Part_11 Hanger* enables the HSI to be mounted on UAVs, conveyor belts, and other devices.
- The *Part_12 Hanger clip* locks the HSI's hanger.
- The *Part_E02 Editable front lens* enables the user to substitute the frontal lens.

Table 1

The performance of the proposed HSI device and the more comparable HSI devices found in the literature. A cell is blank when the parameter is not available in its related paper. $|\lambda|$ is the number of wavebands. The column *One Shot* has a tick mark when the HSI device extracts a hypercube from a single digital image. The *OM* column has a cross mark when the HSI device possesses optomechanical parts or, it uses the whisk-broom technique to acquire hypercubes. The column *SW* has a tick mark when the authors deployed the required software to calibrate and obtain hypercubes.

Device	Cost US\$	Weight g	Size cm	Spectral Range nm	Spatial Resolution px	$ \lambda $	One Shot	OM	SW
Proposed	500	300	$18 \times 9 \times 7$	420–950	116×110	315	X	X	X
2018 [7]	626	306	$15 \times 6 \times 5$	435–733	–	212	X	X	X
2016 [8]	10,000	–	–	340–820	18×1	50	X	X	X
2015 [9]	1,000	1,200	–	340–750	–	30	X	X	X
2012 [10]	2,000	1,500	$15 \times 12 \times 25$	410–700	120×120	54	✓	✓	X
2011 [11]	$\geq 2,000$	–	–	400–1,000	800×1	205	X	X	X
2010 [12]	–	–	–	450–650	285×285	60	✓	✓	X
2008 [13]	–	–	–	460–880	400×400	17	✓	✓	X
2007 [14]	–	950	–	450–700	64×64	50	✓	✓	X

- The *Part_E08 Editable seal extension* enables the user to modify the length of the *Seal extension* if it is needed.
- The *Part_E10 Editable lid* enables the user to substitute the connector of the power supply.

4. Bill of materials

The cost of printing the set of 3D models listed in Section 3.1 was calculated, considering only the raw material.

ID	Component	Qty.	Unit cost	Total cost	Source	Material type
RPI01	Raspberry Pi 3 B + with charger	1	\$54	\$54	canakit.com	Electronics
RPI02	Raspberry Pi NoIR V2 Camera 8 Megapixels	1	\$37	\$37	canakit.com	Electronics
RPI03	Mini SD memory, 32 Gb	1	\$18	\$18	canakit.com	Electronics
PS01	USB cable	1	\$6	\$6	amazon.com	Electronics
PS02	Power Supply Connector	1	\$1	\$1	amazon.com	Electronics
PS03	Power On/Off Switch	1	\$1	\$1	amazon.com	Electronics
PS04	Connector cables	2	\$0.5	\$1	amazon.com	Electronics
LENS01	Macro + 10 lens	1	\$15	\$15	amazon.com	Optics
LENS02	35 mm C Series Fixed Focal Length Lens. Model: 59872	1	\$320	\$320	edmundoptics.com	Optics
LENS03	Double-axis diffraction grating	1	\$1.5	\$1.5	amazon.com	Plastic
3DM01	Part_01 Case	1	\$13	\$13	...	PLA
3DM02	Part_02 Front lens	1	\$3	\$3	...	PLA
3DM03	Part_03 Square aperture	1	\$1	\$1	...	PLA
3DM04	Part_04 50 mm extension	2	\$7	\$14	...	PLA
3DM06	Part_06 Diffraction grating	1	\$1	\$1	...	PLA
3DM07	Part_07 Camera base	1	\$1	\$1	...	PLA
3DM08	Part_08 Seal extension	1	\$2	\$2	...	PLA
3DM09	Part_09 X extension	1	\$2	\$2	...	PLA
3DM10	Part_10 Lid	1	\$3	\$3	...	PLA
3DM11	Part_11 Hanger	1	\$4	\$4	...	PLA
3DM12	Part_12 Hanger clip	1	\$1	\$1	...	PLA
CAL01	Sylvania CFL 2700 K Bulbs	1	\$3	\$3	amazon.com	Fluorescent
CAL02	Filament Incandescent Light Bulb	1	\$11	\$11	amazon.com	Thugsten
CAL03	Work Light Metal Guard	1	\$20	\$20	amazon.com	Metal

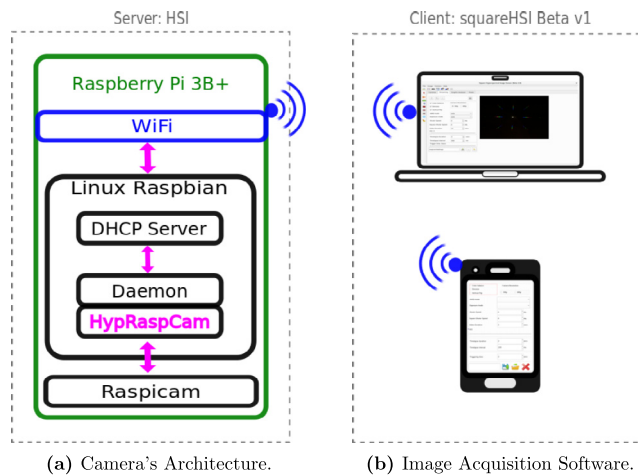


Fig. 2. The proposed system. (a) The HSI works as an image acquisition server. As soon as it is turned on, it executes the program [HypRaspCam](#) to keep it attending image requests from the Client software. (b) The [squareHSIBetaV1](#) (available for Linux PC and Android v7+ devices) acquires images using the services of the Server. Afterward, the Client software builds and exports hypercubes.

5. Build instructions

5.1. System overview

The proposed HSI system is composed of two wirelessly connected devices:

1. The proposed HSI device is an image acquisition *Server*¹. It has the [HypRaspCam](#) software installed in it ([Video tutorial](#)). Furthermore, the HSI device creates a WiFi network immediately after it is turned on, and it keeps providing services while it is turned on.
2. The *Client*² which is a device with *Linux*³ or *Android*⁴ (V7+ and 5-inch screen). This device is enabled when the image acquisition software ([squareHSIBetaV1](#)) is installed in it.

5.2. Software installation

Together with the proposed HSI device, a stack of software is provided. It must be installed before to use the HSI system. We have created a set of video tutorials with instructions to install this software. Those tutorials are available in the following list:

1. How to install the Operative System (OS) on an RPI ([Video tutorial](#)).
2. How to enable the Raspicam in the OS ([Video tutorial](#)).
3. How to install the image acquisition *Server* ([Video tutorial](#)).
4. How to install the image acquisition *Client* ([Video tutorial](#)).

5.3. Building the front lens

The front lens allows to focussing the examined object onto the imaging sensor inside the proposed HSI device as well as it avoids dust. Given that handle the optics without protection, you may contaminate it. Therefore, we recommend you to wear cloth gloves during this procedure, which is simple but essential. The aim is to immobilize the lens (Edmundoptics Model 59872) when it is attached to its mount ([Part_02 Front lens](#)). The lens mount has a small hole on the side to lock the lens with its screw ([Fig. 3](#)). If you want to substitute the proposed lens, then, you need to modify the lens' mount and adapt it to your preferred lens. In such case, you can edit the source file: [Part_E02 Editable front lens](#).

¹ A server is a device that provides functionality for other programs or devices, called "Clients".

² A client is a device or a program that accesses a service made available by a server.

³ Linux is an operating system that manages all of the hardware resources associated with your desktop or laptop.

⁴ Android is a mobile operating system developed by Google.

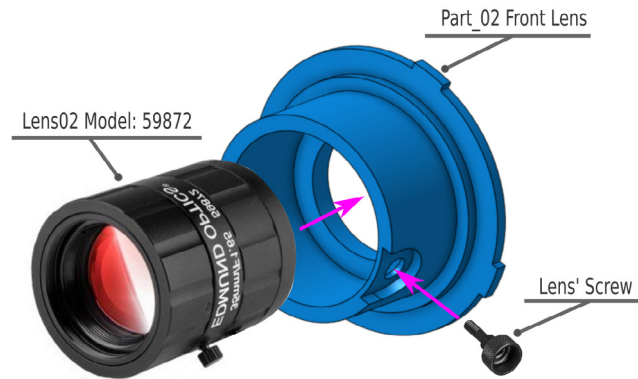


Fig. 3. Figure with instructions to attach a frontal lens to the [Part_02 Front lens](#).

5.4. Building the diffraction grating

Next, you need to prepare a double-axis diffraction grating with 13,500 lines/inch (Fig. 4). The recommended method to do it is the following:

1. You have to wear cloth gloves or some protection because the diffraction grating is very fragile.
2. You have to cut a concentric rectangle from the diffraction grating.
3. That rectangle must have the size of its mount's frame ([Part_06 diffraction grating](#)).
4. Carefully, you have put together the diffraction grating onto its mount. You have to immobilize it with paper tape, avoiding to cover the transparent plastic of the diffraction grating.

5.5. External power supply cable

The RPI inside the proposed HSI device is a computer (CPU) of small dimensions. This CPU consumes +5 volts (VCC). In order to measure +5 volts, it is required a reference voltage. This reference voltage is called ground (GND). Therefore, the order of the VCC and GND wires is strict. Otherwise, the proposed HSI device will not turn on.

The proposed HSI device uses a USB cord as external power supply cable because it is compatible with many available devices, chargers, and connectors. Furthermore, a USB cord is convenient because the VCC and GND wires are very easy to identify. It has four wires, each of them has a different color: Red (VCC), black (GND), white (positive data), and green (negative data). You have to isolate both data wires because the proposed HSI device does not use them. Furthermore, you have to solder the VCC and GND wires to the 2-pin male connector (Fig. 5).

5.6. Power supply

Now, we need to prepare the connection between the external power supply and the RPI. To do it correctly, the inner side of the lid (Fig. 6a) has a symbol "+" to indicate the VCC input pin location. Furthermore, it has two holes, one is reserved for a 2-pin connector 2510, and another is reserved for an ON/OFF switch (a 2-pin snap-in button rocker switch). The red line represents the VCC, and the black line represents the GND (Fig. 6). Both the red and black wires were soldered to a micro jst 2.0 Ph 2-pin connector.

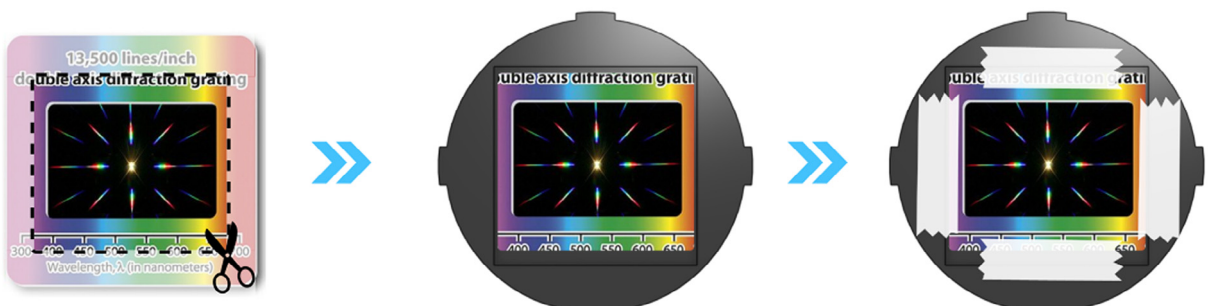


Fig. 4. Figure with the method to prepare a diffraction grating.



Fig. 5. External power supply cable made with a USB cord.

Moreover, the RPI has a General-Purpose Input/Output (GPIO) (Fig. 6b). The fourth and sixth pines of the RPI's GPIO correspond to the VCC and GND inputs, respectively. We have to solder those pines to a female micro jst 2.0 Ph 2-pin connector.

6. Assembling an HSI device

For the following procedure, we assumed that you had printed the set of 3D models in a thermoplastic 3D printer as well as the optics and electronics are ready. In such case, you can assemble the proposed HSI device (Video tutorial). Furthermore, the order you have to introduce those components into the HSI's case is shown in Fig. 7.

7. Calibration

Once you have assembled the proposed HSI device, it is ready to take photographs (Video tutorial). They are also known as *Spectral Photographs* (SP), because they contain diffracted light. These SPs have two distinguishable areas: The zero-order mode (ZM) and the first-order mode [10]. The ZM occurs when the light is not diffracted. It appears as a small square, located at the center. Furthermore, the first-order mode is the first time that the light is diffracted. It has a higher intensity than higher modes. Moreover, its perimeter contains the ZM. Hence, this area will contain the mayor amount of information and energy. Therefore, the first-order mode is the *Region of Interest* (ROI).

The identification of the ZM, ROI, and other references enables the Client software to calculate the parameters it needs to export HIs. This process is also known as *calibration*. Unlike digital cameras, this calibration involves the spectral properties of the diffraction system in the proposed HSI device. It does not considers the variation of light which is usually mitigated using a normalization based on *White* and *Dark* references [16]. Thus the normalization is a process applied to HIs, and the calibration is a requirement to generate them.

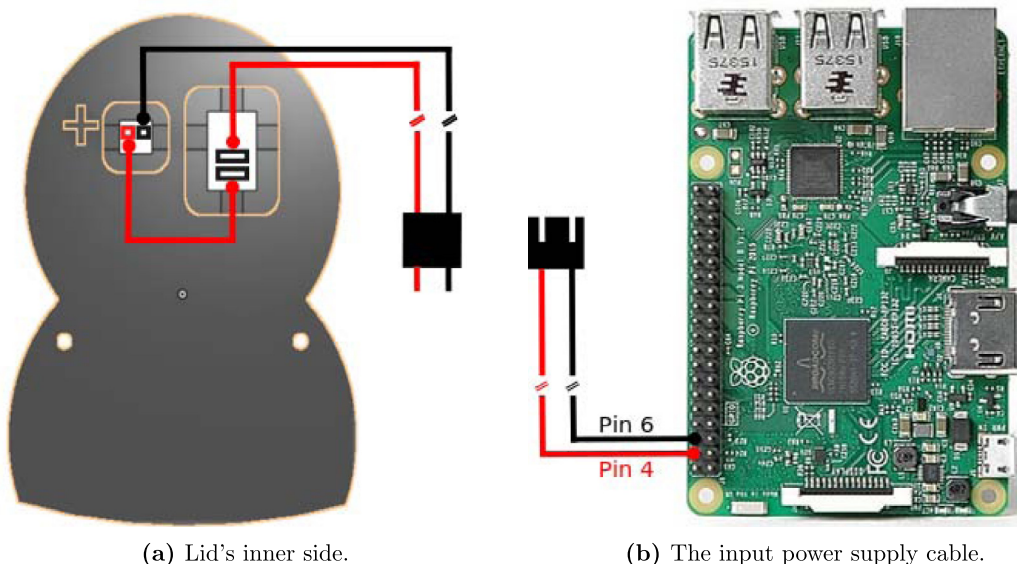


Fig. 6. The connection between the external power cable and the Raspberry Pi 3B + inside the proposed HSI device.

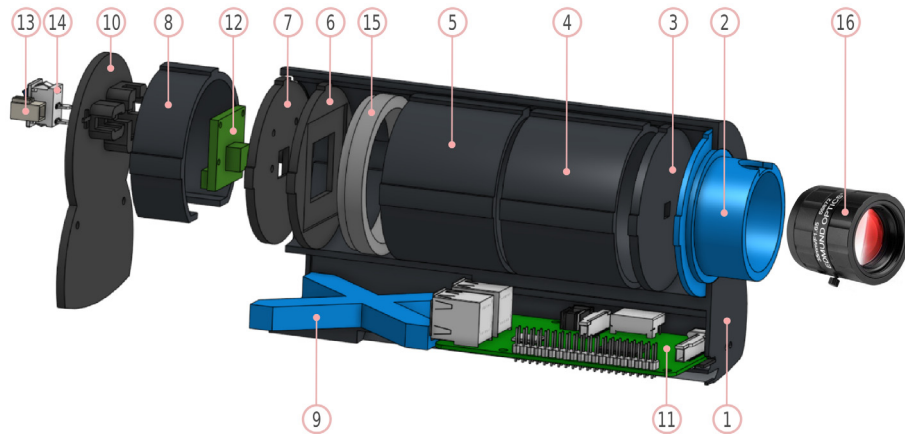



Fig. 7. The HSI's components. 1) Part_01 Case. 2) Part_02 Front lens. 3) Part_03 Square aperture. 4) Part_04 Extension 1 of 2. 5) Part_05 Extension 2 of 2. 6) Part_06 Diffraction grating. 7) Part_07 Camera base. 8) Part_08 Seal extension. 9) Part_09 X extension. 10) Part_10 Lid. 11) RPI01 Raspberry Pi 3 B+. 12) RPI02 Raspberry Pi NoIR V2 Camera 8 Megapixels. 13) PS02 Power Supply Connector. 14) PS03 Power On/Off Switch. 15) +10 Macro 52 mm Lens. 16) LENS02 35 mm C Series. Edmundoptics, model 59872.

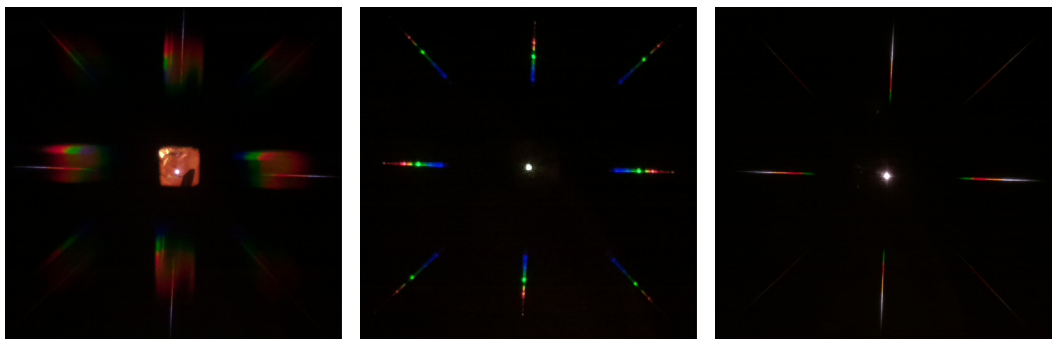
7.1. Reference images

To calibrate the proposed HSI device, you have to take three SPs (Fig. 8). They have to contain information to allow you to identify the location of the spectral references. The perimeters of the ROI and ZM have to be identifiable in the first SP (Fig. 8a). The second SP have to contain a diffracted light with three or more well known wavelength peaks. We used the light of a fluorescent lamp *Sylvania Delux EL 2700 K spotlight* (Fig. 8b) because it has five identifiable wavelength peaks at 405, 440, 545, 613, and 710 nm. The third SP has to have a diffracted light with a continues spectrum. We used an incandescent lamp because it has a light spectrum with energy in whole operation range of the Raspicam (Fig. 8c). Furthermore, the origin of the light of the fluorescent and incandescent lamp have to be identifiable. Such that you could distinguish its center. With the coordinates of the light's origin and the coordinates of the diffracted wavelength peaks, the Client software is able to calculate the relationship between wavelength-to-pixels and vice versa. Afterward, the Client software uses this relationship and the continues spectrum of light (Fig. 8c) to calculate the independent performance of the red, green and blue sensors of the imaging sensor inside the proposed HSI device.

Due to the source of light have to be as small as possible, we covered the lamps with two layers of aluminum foil. Avoiding any output of light. Next, we punctured the aluminum foil using a thin needle. It allows to pass a very small amount of light. Thus we placed the HSI device 40 cm away from that tiny source of light (Fig. 9). Because the lamp is complete covered, this method generates much heat. Therefore, you have to perform this procedure carefully. An alternative is to use a metal box with a small hole. We extended this instructions in the following [video tutorial](#).

7.2. Rectangular references

The first reference you need to define is the perimeter of the ROI ([Video tutorial](#)). You can use the tool  to draw it on the first reference SP (Fig. 8a). Next, do double click over that rectangle to pop-up the menu (Fig. 10a). Thus you can save it as the



(a) Region of Interest.

(b) The diffracted fluorescent light.

(c) The diffracted incandescent light.

Fig. 8. The set of reference images.

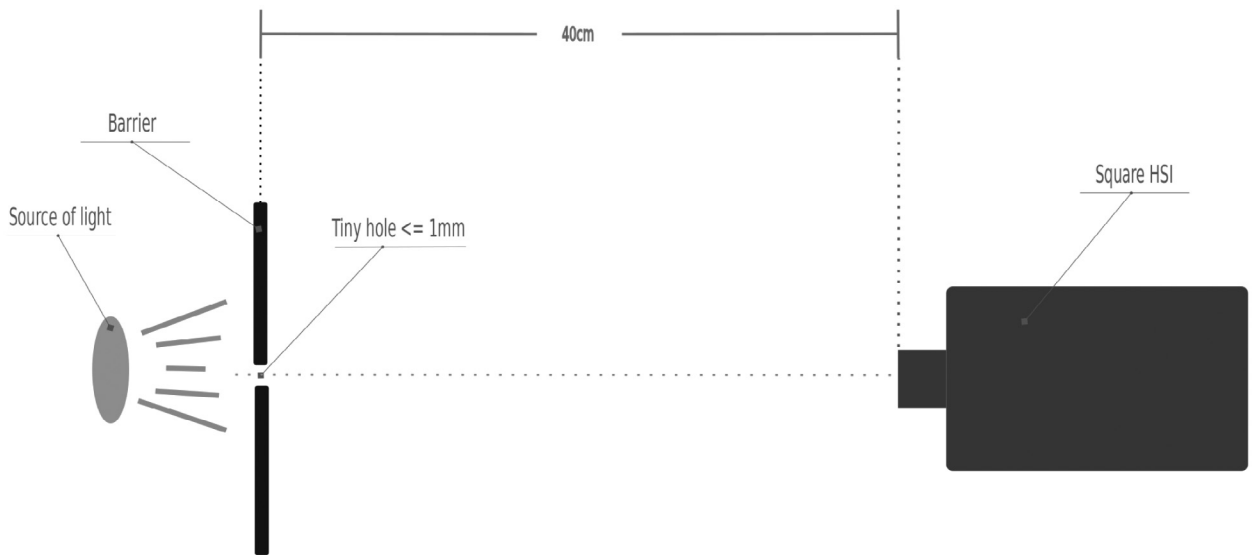
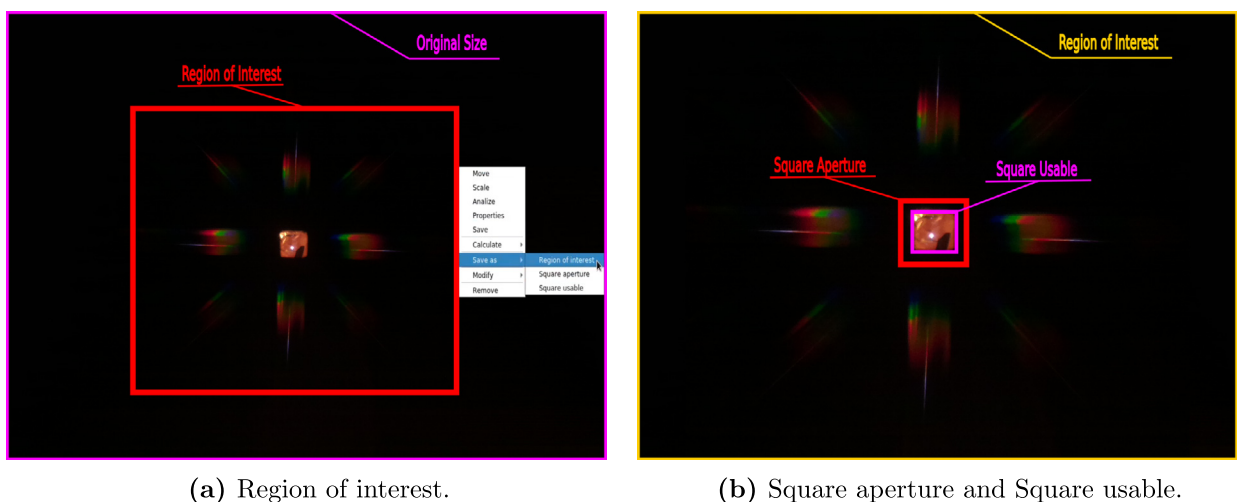


Fig. 9. A barrier with a tiny hole blocks the source of light. The HSI and the source of light are aligned. The HSI device is placed 40 cm away from the source of light.

ROI with the following menu path: *Save as* → *Region of Interest*. This is the only calibration reference to be created in the original-size spectral photograph.

Once you have created the ROI reference, you can delete the pixels outside it with the following menu path: *Main menu* → *Camera* → *Apply Region of Interest*. Thus the resultant sub-image will have the pixels inside the perimeter of the ROI only. Moreover, the ROI reference enables the Client software to transmit that sub-image only, and then, it minimizes the image transmission time. *You have to create the remainder references using the ROI sub-image*. For example, the *Square aperture* and the *Square usable* (Fig. 10b). The method to define and save the perimeter of those areas is very similar to the method used to create the ROI. The difference between those methods is the option you will choose in the menu ‘*Save as*’. If the rectangle that you are saving is not the ROI, then, click the on the option *Square aperture* or *Square usable* as it corresponds.



(a) Region of interest.

(b) Square aperture and Square usable.

Fig. 10. (a) The *Region of interest* (Red rectangle) should be specified using the original size spectral photograph (Magenta frame). (b) We have cropped the *Region of interest* (Yellow frame) from the original-size spectral photograph, and then, the resulting image was used to specify the *Square aperture* (Red rectangle) and the *Square usable* (Magenta rectangle). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

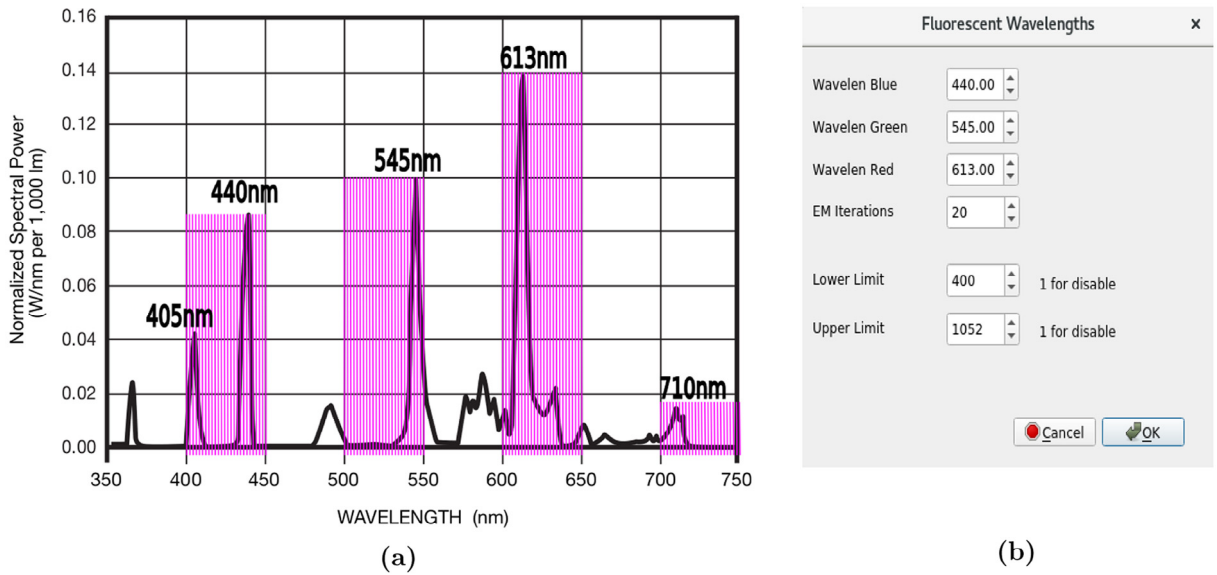



Fig. 11. (a) The light spectrum of a *Sylvania Delux EL 2700 K* spotlight. The magenta lines are placed every 2 nm in order to identify the peak wavelength locations. It was redrawn from www.sylvania.com. (b) The window of the fluorescent settings. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

7.3. Spectral references

Next, you have to identify the location of three wavelength peaks in the four diffraction directions, one for each color sensor (red, green, and blue) as well as configure the Client software with their corresponding wavelengths ([Video tutorial](#)). The tool  opens the form that allows you to do it ([Fig. 11b](#)). We choose the 440, 545, and 613 nm peaks ([Fig. 11a](#)) because they are easy to identify in our reference SP ([Fig. 8b](#)). This form also allows you to modify the parameter *EM Iterations* which increases the accuracy of the exported hypercube [10]. Like other authors [10,17], we have concluded that it converges in up to 20 iterations. Furthermore, you can modify the *Lower limit* and the *Upper limit*. These parameters enable you to fix

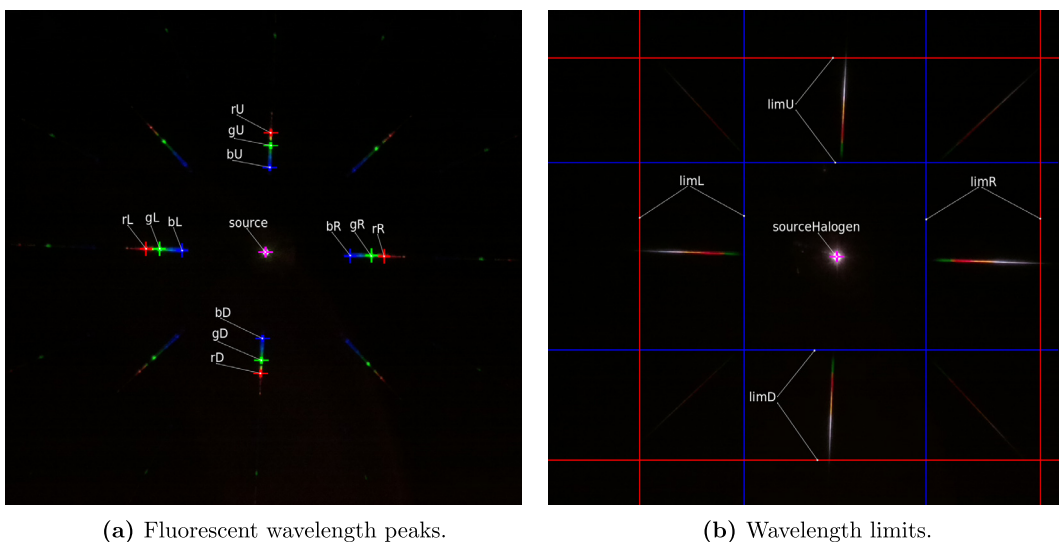
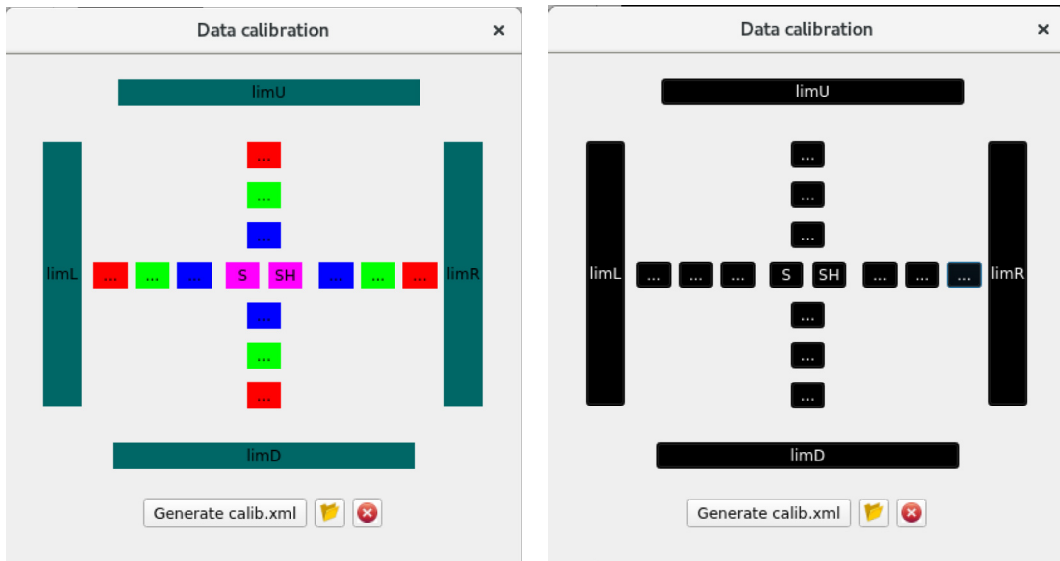


Fig. 12. (a) The red, green and blue crosses identify the 440 nm, 545 nm and 613 nm wavelength peaks of a *Fluorescent Sylvania Delux EL 2700 K*, the magenta cross is the center of the zero-order diffraction, the white text next to each cross is the mandatory filename of each calibration reference. (b) It is a diffracted light spectrum of an incandescent lamp. The red lines are the longest wavelength boundaries, as well as the blue lines are the shortest wavelength boundaries, the magenta cross is centered on its zero-order diffraction. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



(a) Undetected calibration references.

(b) Detected calibration references.

Fig. 13. Data calibration form, before and after the user has generated the reference calibration files. The black color buttons indicate when the calibration software has been detected in the corresponding reference file.

the minimum and maximum wavelength in the exported HI. If they are set to 1, the Client software calculates the operation range. In such case, you will have to define the operation limits by visual inspection of the Fig. 8c (Video tutorial).

The output of this process is shown in Fig. 12. The text at the side of each drawn mark is the filename that you must use to save each of those calibration references (Video tutorial).

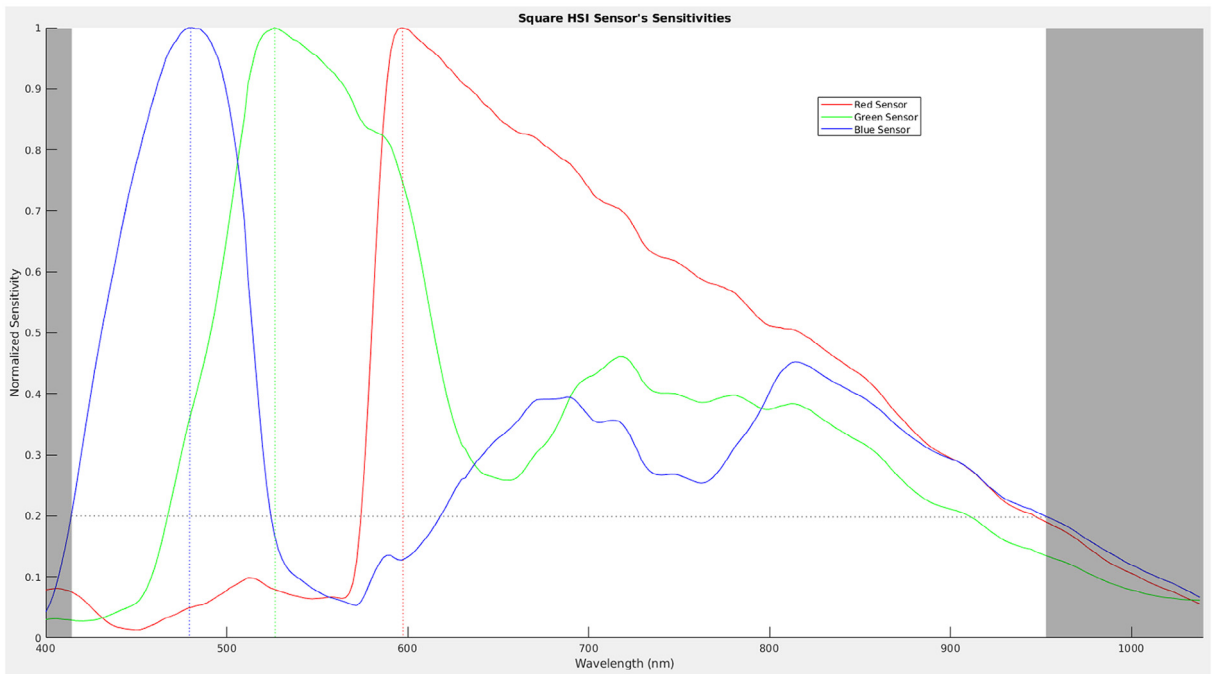


Fig. 14. The sensitivities of the proposed HSI's imaging sensor.

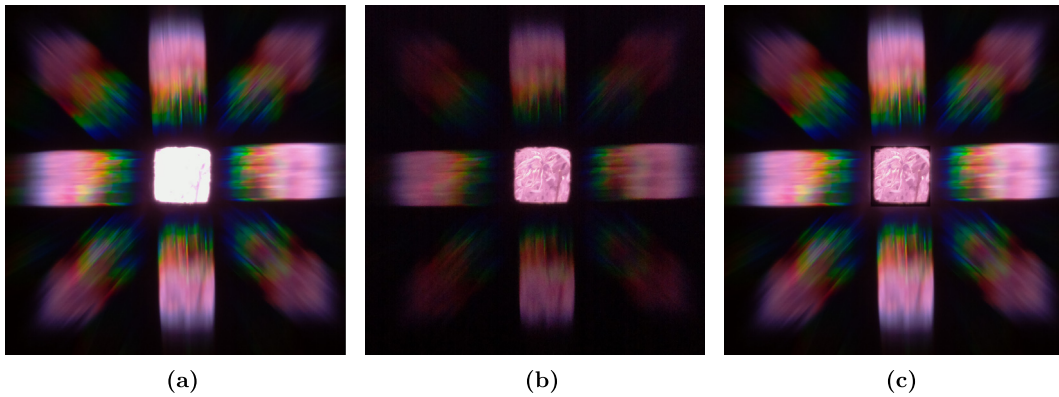


Fig. 15. It shows the three possible light-balance scenarios. (a) An SP with a light-saturated ZM area. (b) An SP with a low-intensity diffracted light. (c) A light-balanced SP.

7.4. Saving calibration

The calibration process finishes when you generate a calibration file. To do it, it is necessary to have all the references previously created. Therefore, you can inspect the status of the references with the tool `hycalib`. This tool uses a color code to express if a reference has been detected or not. When the calibration software detects a calibration reference file, then, it colors the corresponding button in black (Fig. 13b), otherwise it appears in a different color (Fig. 13a).

If the Client software detects every calibration reference, then, you are able to finish the calibration procedure by clicking the button `Generate calib.xml`. Thus it exports the following calibration file: `[INSTALLATION PATH]/XML/hypcalib.xml`.

That file contains the calibration parameters as well as a script to plot the normalized sensitivities of the blue, green and red sensors of the Raspicam inside the proposed HSI device. This script runs either in Octave [18] or Matlab [19]. We executed the script in Octave 4.2.2 to obtain the plot in Fig. 14.

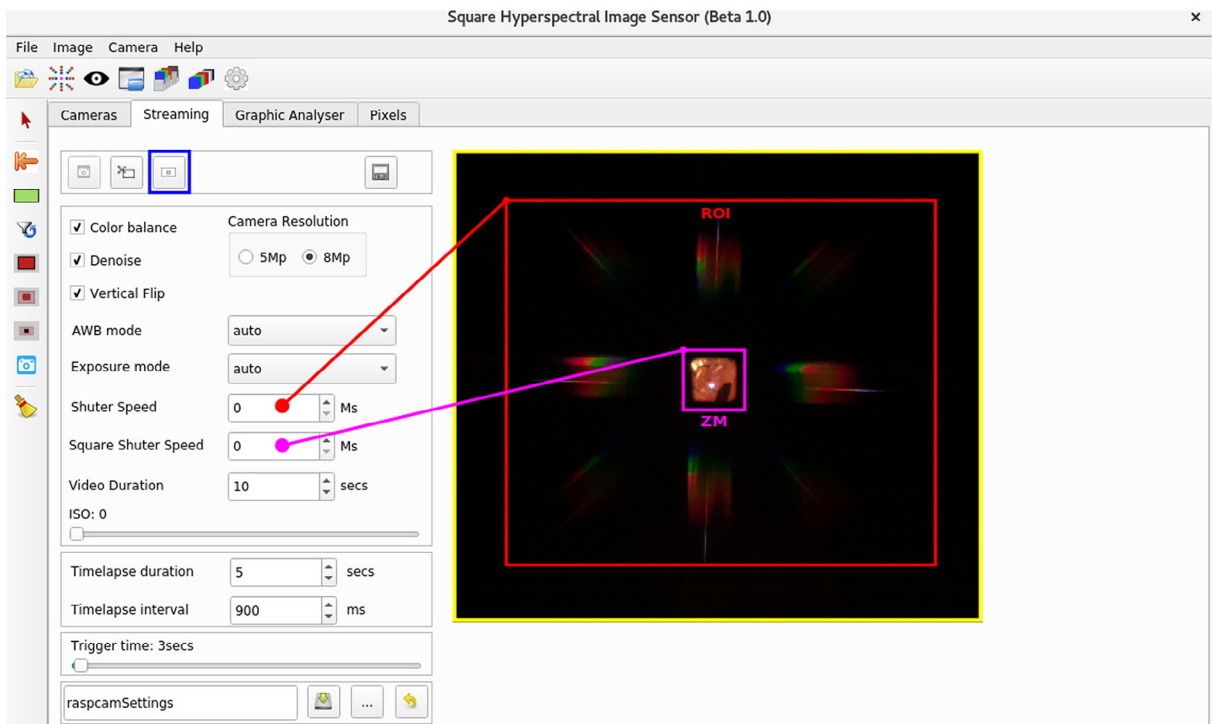


Fig. 16. It shows the ZM and ROI boundary areas. The remarked button allows taking an image with two different exposure times.

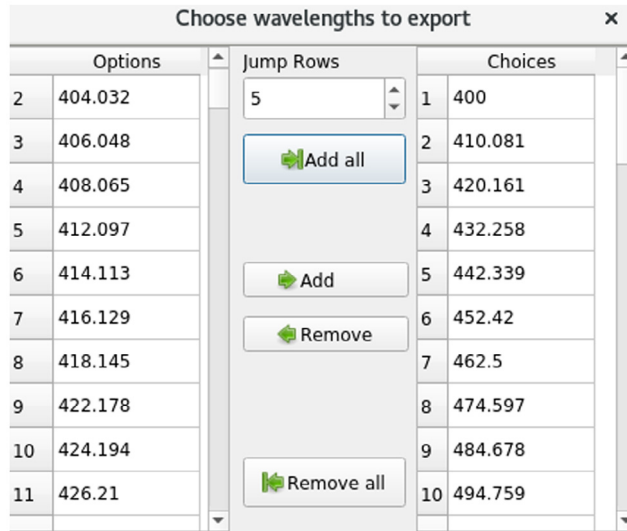


Fig. 17. “Choose wavelengths to export form”. The left table shows every available wavelength that may be included in the exported HI. Furthermore, the right table shows the chosen wavelengths.

7.5. Validating the calibration

An important advantage of the reported HSI system is that it has different tools to validate the calibration (Video tutorial). One of them was used in Section 7.4 to obtain the status of the calibration references. Another tool is the script in the calibration file. An expected plot will show the sensors with high sensibility in the range 420–520 nm (blue), 500–610 nm

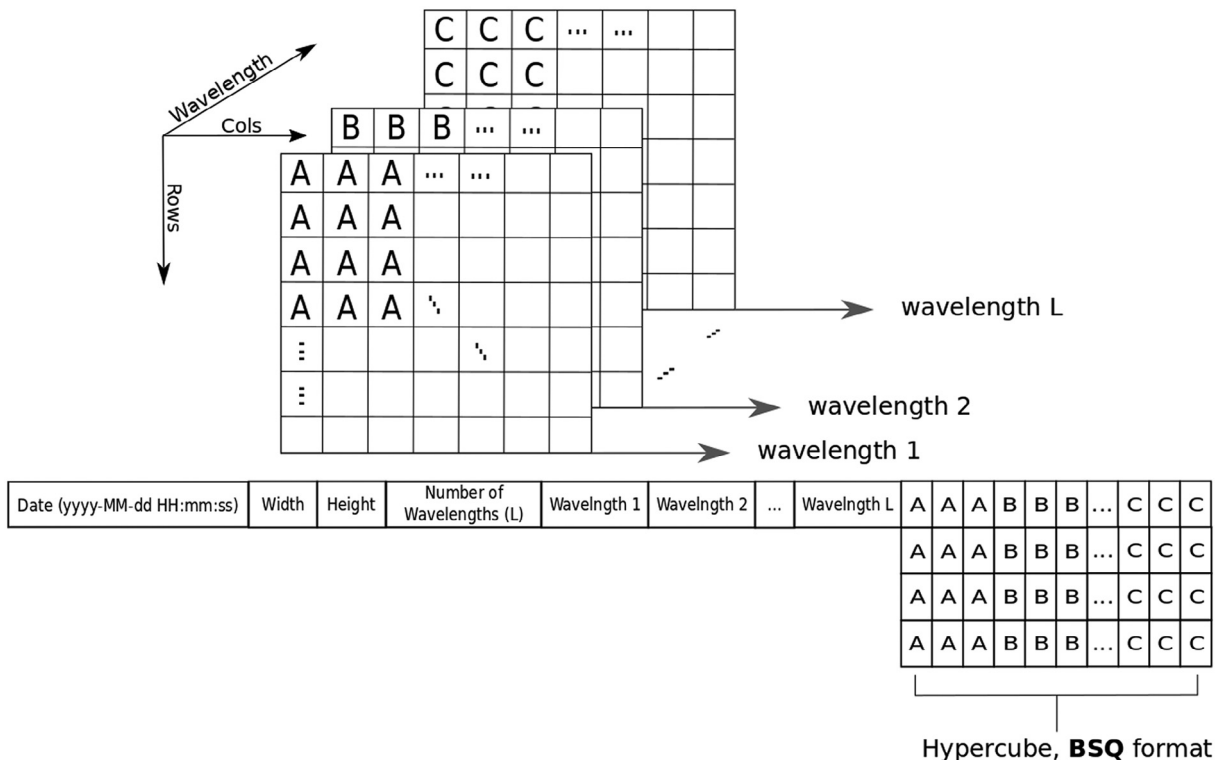


Fig. 18. Band-Sequential (BSQ) redrawn from Matlab [19]. The lower string describes the encoding scheme that the Client software uses to export a HI.

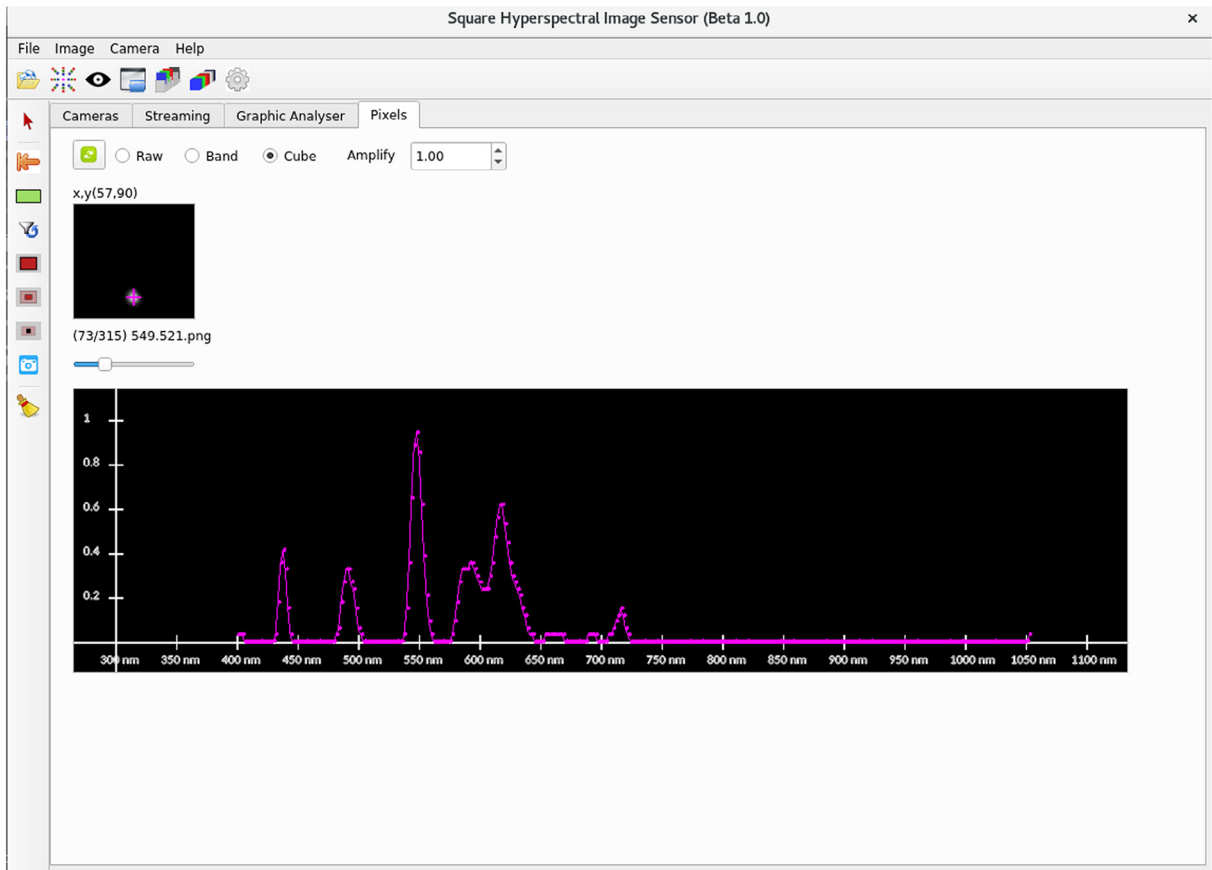



Fig. 19. It displays the “Pixels Tab” of the Client software. It has tools to perform a preliminary analysis of any exported HI. The upper-left image is the 73-th waveband in a HI with 315 wavebands. The HI contains the light spectrum of a *Fluorescent Sylvania CFL 2700 K Bulb*. The mouse pointer is located over the pixel at the 57-th column of the 90-th row in the HI (Magenta cross). The values of the selected pixel are potted in the lower graph.

(green), 580–720 nm (red), and 790–850 nm (blue). If your result is inconsistent with this, then, it is probable that your calibration is incorrect.

Furthermore, the tool  allows you to check the created references and its calculated locations. It allows you to validate the detected rotation, specified limits, calculated limits, specified centroids, and calculated centroids. As well as observe the expected location of any waveband in the spectral operation range of the proposed HSI device. With this tool, the Client software allows you to evaluate every phase of the calibration process. Hence, you will be able to calibrate the proposed HSI device in a very accurate way, investing a short time.

8. Exporting Hyperspectral Images

Once you have calibrated the proposed HSI device, then, it can reconstruct HIs from an SP by inverting its diffraction projection with the *Expectation-Maximization* (EM) algorithm [10]. We implemented and embedded this algorithm in the Client software, such that, it drastically decreases your processing time. The Client software requires two inputs to export a HI: A light-balanced SP, and the set of wavelengths you want to include in the HI.

8.1. Taking a Light-balanced SP

The proposed HSI device is based on the *Computed Tomographic Imaging Spectrometer* (CTIS) technology [10,20,17]. These type of devices have much light in the ZM area about its diffracted light. This characteristic may produce SPs with unbalanced luminance. For example, a light-saturated ZM area (Fig. 15a), or a low-intensity diffracted light (Fig. 15b). However, they need a light-balanced SP to reconstruct a HI (Fig. 15c). A light-balanced SP has similar brightness in the ZM and the diffracted light areas, as well as it allows you to distinguish the sensed object. Furthermore, the diffracted light is bright, and it is not light-saturated.

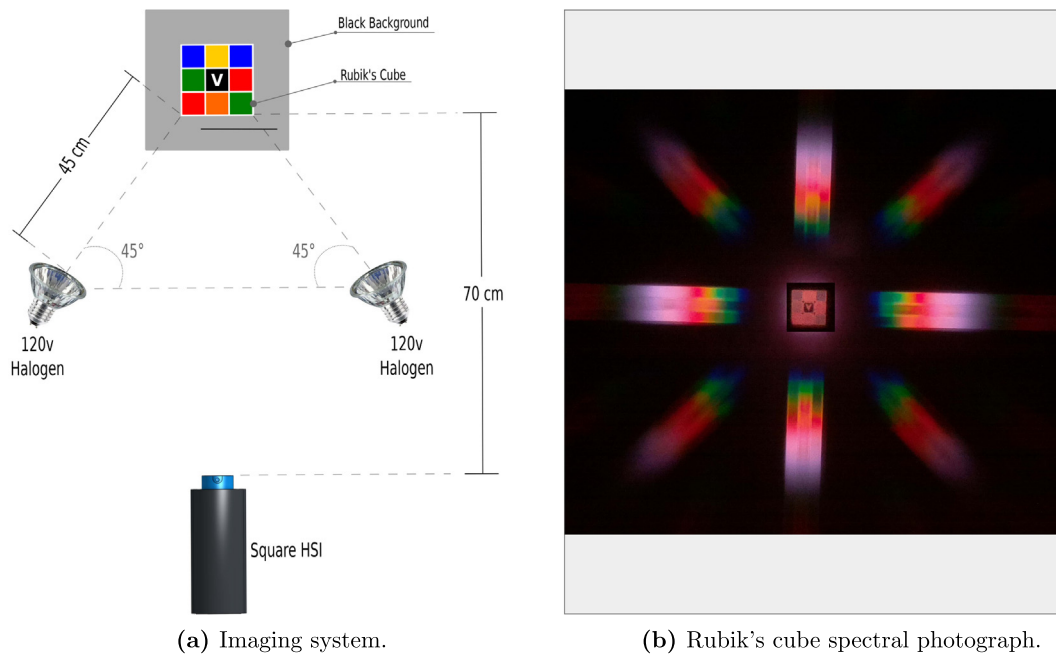




Fig. 20. (a) It shows an imaging system which includes the proposed HSI device. This imaging system was used to acquire a HI from the face of a 3×3 V-CUBE® Rubik's cube. We prepared a face of the Rubik's cube with five different colors (Orange, Green, Black, Blue, and Red). Furthermore, we placed a black background to increase the quality of the taken SP. Moreover, the illumination system was composed of two 120v halogen lamps. These lamps were placed 45 cm away from the Rubik's cube at an angle of 45° . (b) An SP of the Rubik's cube. It was taken with the described imaging system. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Unlike previous works [10,20,17], the proposed HSI device avoids this drawback because it allows you to take SPs with either same or different exposure times for each of those areas. Besides, the Client software has the tool “Main menu → Image → Extract aperture” that allows you to merge two SPs into one. We used this tool to combine the diffracted area in Fig. 15a and the ZM in Fig. 15b into Fig. 15c, which is the expected light-balanced SP. .


Moreover, the tool  allows you to take a light-balanced SP with two exposure times. One for the ZM and another for the ROI. We highlighted these areas with a red and a magenta rectangle in Fig. 16. Besides, we linked each of them to a spin-box that sets the exposure time of the corresponding boundary area. When you set those spin-boxes to zero, the software will then take SPs with automatic exposure times.

8.2. Selecting wavelengths

The proposed HSI device allows you to choose the set of wavelengths to be included in a HI before to export it. Because you can select all the prospect wavebands or a subset of them, you will be able to perform experiments and preliminary analysis very quickly. The Client software automatically obtains the list with options during the calibration process (Section 7).

The tool  opens the form where you can select the list of wavelengths to be include in a HI (Fig. 17). When you set the spin-box “Jump Rows” to 1, and you click the button “Add all”, the Client software adds every available wavelength in the “Options table”. Furthermore, if you set it with a number k greater than 1, then, the Client software adds the wavelength in the 1st row, k -th row, 2 k -th row, and the rest until the last row (Video tutorial).

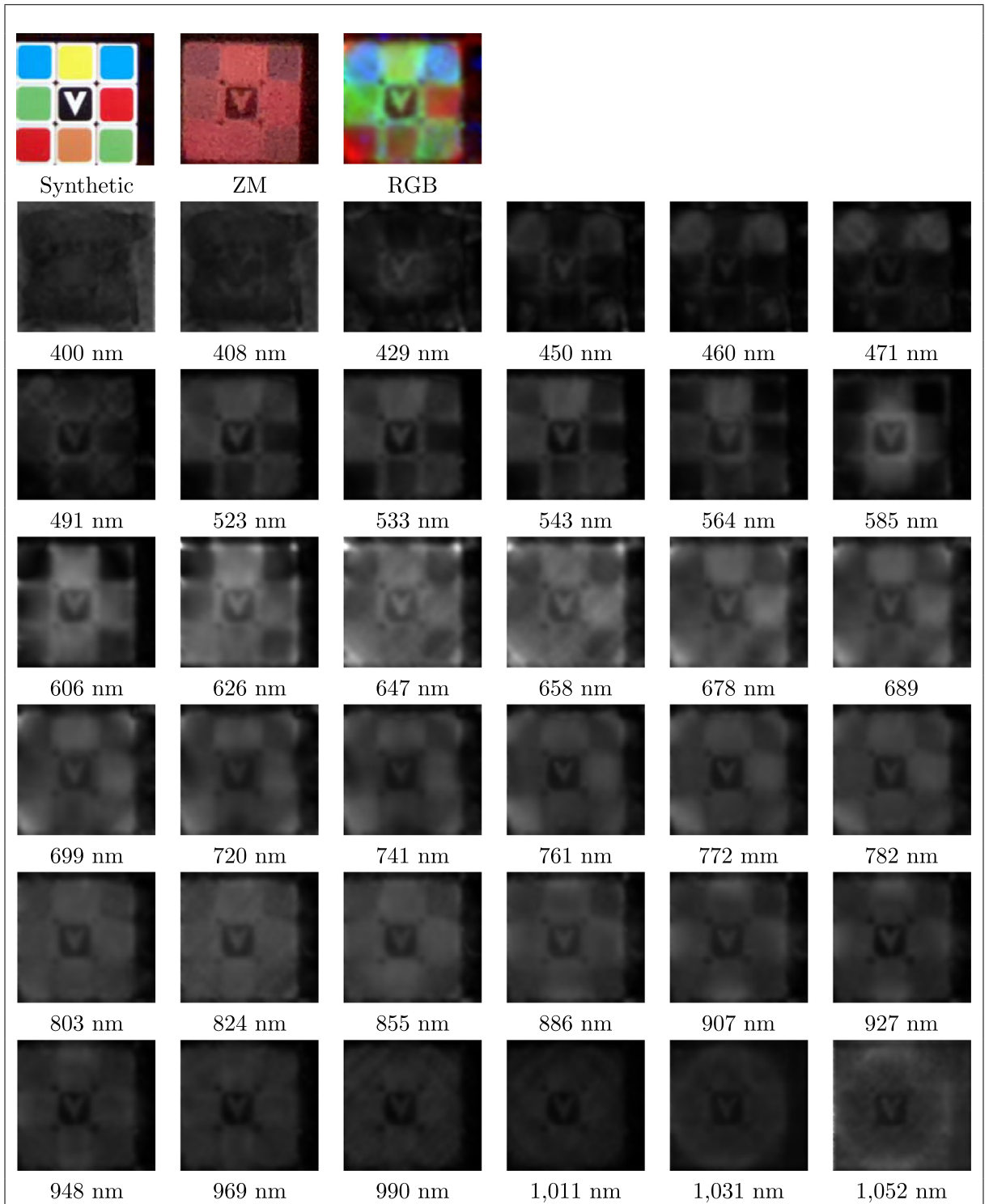
8.3. Exporting Hyperspectral Images

Once you have taken a light-balanced SP (Section 8.1) and you have selected the set of wavelengths to be included in the HI (Section 8.2), the tool  allows you to export a HI. This tool will ask you for the output filename as any other *Linux application*. Afterward, it will generate a file in the selected directory. This file will be saved with the extension “.hypercube”. The Client software exports a HI in the Comma Separated Values (CSV) format with the HI's meta-data embedded in it. This format is convenient because it is a standard and portable format. Thus it enables you to analyze the exported hypercubes with any other software, for example, Octave or Matlab.

The proposed HSI device exports a HI with the following Encoding Scheme: [Date and time of creation], [Number of columns (width)], [Number of rows (height)], [Number of wavelengths (L)], [L wavelengths values (nanometers as real numbers)], [The spectral pixels values encoded using Band-Sequential (BSQ)] (Fig. 18).

Table 2

(*Synthetic*) It is a digitally created image. We have taken it with a cell phone camera. Afterward, we made it match the dimensions and alignment of the ZM. (*ZM*) It is the real “Square Aperture” boundary area. (*RGB*) We made it by merging the slide images at 678, 523, and 471 nm as red, green, and blue colors respectively. (400 nm–1052 nm) It is a subset of 36 wavebands, interleaved selected, and 6x scaled for clarity.



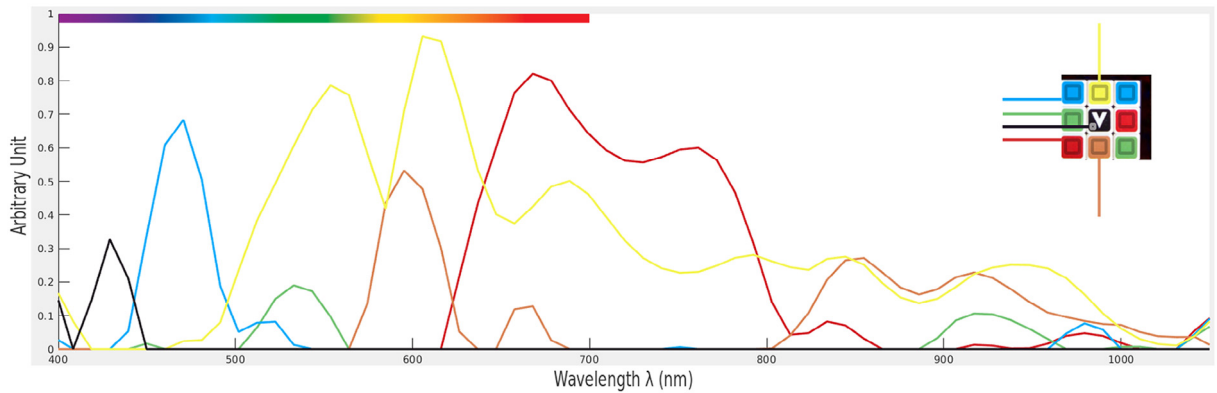


Fig. 21. It shows the normalized mean spectrum of the color squares in a V-CUBE® Rubik's cube (red, green, blue, orange, yellow, and black). We subtracted the mean spectrum of the entire Rubik's cube from the mean spectrum of every color, and we bounded them to non-negative values before to be plotted. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The process to export a HI finishes when the Client software automatically imports the just created HI. Because you have the format of the exported HI (Fig. 18), you can open and analyze it in your preferred software. Furthermore, the tool allows you to load a HI previously created with the Client software.

When you load a HI, the Client software automatically extracts every waveband image and save it in the folder: "Installation path → tmpHypCubes". Afterward, it displays the "Pixels Tab" which has tools to perform a preliminary analysis (Fig. 19). This tab has a horizontal slider that allows you to change the displayed waveband image. Furthermore, when you pass the mouse over the current waveband image, the Client software plots the values of the selected pixel. Thus you could perform a preliminary analysis of any previously exported HI.

9. Validation and characterization

9.1. Characterization

Based on Fig. 14, the proposed HSI device detects the spectrum of light from 400 to 1050 nm. It has four peaks of sensitivity at 481, 526, 597, and 820 nm. Furthermore, it has a low sensitivity for shorter wavelengths than 415 nm and longer than 950 nm. With this information, you can manually configure the Client software with the more convenient shortest and longest wavelength to be included in the exported HI. Besides, the proposed HSI device uses parts printed in a thermoplastic 3D printer. In consequence, its maximum operating temperature is 45 °C. Moreover, we have operated the proposed HSI device in outdoor environments with temperatures from −20 °C to 35 °C.

9.2. Validation

The following multi-purpose examples aim to show the performance of the proposed device under working conditions. One experiment was done under controlled light conditions, and another was done in a landscape.

9.2.1. Rubik's cube

This experiment aims to obtain a HI and use it to analyze the HSI's spectral accuracy. We acquired a HI from the side of a 3 × 3 V-CUBE® Rubik's cube. This Rubik's cube could be a reference because it is available online. Thus anybody can repeat the proposed experiment.

We prepared a side of the Rubik's cube with two squares of every available color. The only exception is the orange color because it appears only once (Fig. 20a). Furthermore, the illumination system is composed of two 120v halogen lamps. These lamps were located 45 cm away from the Rubik's cube at an angle of 45°. Besides, we placed black cardboard behind the Rubik's cube as background. Finally, we positioned the proposed HSI device 70 cm away from the face of the Rubik's cube.

Once the scenario was ready, we acquired a light-balanced SP by setting the Client software with the *Shutter Speed* and the *Square Shutter Speed* parameters to 2000 μs and 50 μs respectively. It was evident the quality improvement obtained with two exposure times in comparison with single exposure time.

Afterward, we configured the "Jump Row = 5" (Fig. 17), and the "EM Iterations = 999" (Fig. 11b). Next, we used the taken SP (Fig. 20b) to export a HI with 64 different wavebands from 400 to 1052 nm, and a spectral resolution of 10.1875 nm. For clarity, we only show 36 of those 64 wavebands, interleaved selected, and 6 × scaled (Table 2). Moreover, we included the following three reference images:

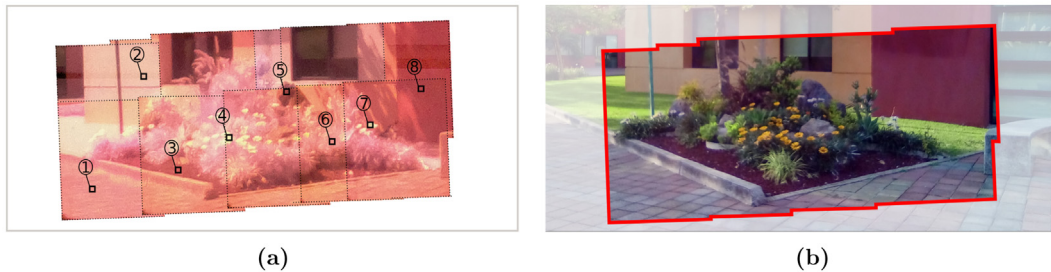


Fig. 22. (a) A set of stitched HIs. The dashed lines identify the perimeter of every stitched HI. It has eight different numbered pixels; each of them corresponds to a different object in the target. (b) For clarity, we have drawn the perimeter of the mosaic onto an RGB image.

1. Synthetic: This image was digitally created. We photographed the Rubik's cube with an RGB camera and edited to make it looks similar to the ZM.
2. ZM: This image is the real "Square Aperture" boundary area. You can obtain it with the tool "Main menu → Image → Extract Aperture". Its low-quality may be caused by the texture of the double-axis diffraction grating in the optical system. Furthermore, it looks different to the "Synthetic" image because the imaging sensor does not have an infrared filter.
3. RGB: It was created with the tool "Main menu → Image → Merge into RGB". We selected the 678, 523, and 471 nm as red, green, and blue colors respectively. Its similarity with the expected image (Synthetic) validates the spectral and spatial performance of the proposed HSI device.

Once exported the HI ([2000_50.hypercube](#)) in the Client software, we opened it in Octave using the script [importHypercube.m](#), which also exports a ".png" image from every waveband in the HI ([Video tutorial](#)). Next, we obtained the normalized mean spectrum of every color square in the face of the Rubik's cube ([Fig. 21](#)) as well as the entire Rubik's cube mean spectrum. Before to plot every color spectrum, we subtracted entire Rubik's cube mean spectrum to them, and we bounded them to non-negative values. Finally, we used Octave to save the original HI in an Octave/Matlab file ([2000_50.mat](#)).

The left-top area in [Fig. 21](#) shows an approximation of the visible light spectrum. Every mean spectrum has been calculated using a concentric square of 20x20 pixels. When the same color appears in two different color squares, then, its mean spectrum is the mean of both color squares. As an exception, the mean spectrum of the black color was calculated using a square of 7x7 pixels which is located at the left-bottom area of the black color. We do it to avoid considering those pixels belonging to its white symbol "V". The spectral accuracy achieved with the proposed HSI device is remarkable.

9.2.2. Landscape

This experiment was conducted with the proposed HSI device in an outdoor scenario. It aims to demonstrate the performance of the proposed HSI device with ambient light conditions. The analyzed target is a landscape which contains many different objects (Stones, flowers, walls, and others).

We placed the proposed HSI device five meters away from the target. This separation was not enough to cover it with a single shot. Therefore, we created a high *spatial-resolution*⁵ HI with the *Image Stitching Technique*⁶ ([Fig. 22a](#)). Furthermore, the perimeter of the resultant high-resolution HI was drawn over an RGB image for clarity ([Fig. 22b](#)).

The spectral analysis is done with eight pixels of the stitched HI ([Fig. 22a](#)). Each of those pixels senses a different object in the landscape. Therefore, we separated them into three different subsets: walls, plants, and stones. This partition was created to compare the obtained *Spectral Signature*⁷ (SS) of similar materials. The RGB color of the corresponding object was not considered in the creation of those subsets.

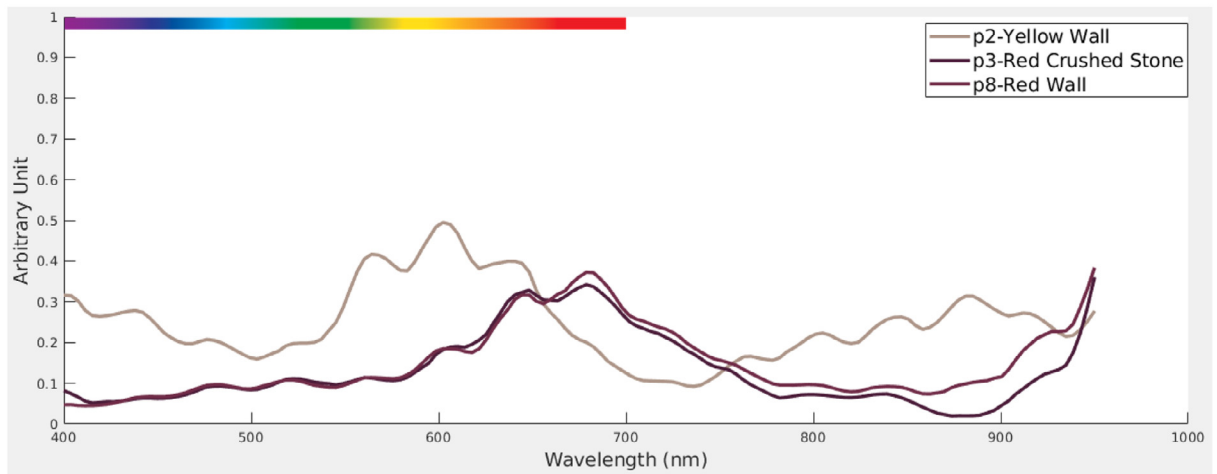
Next, we created a graph for each subset ([Fig. 2](#)). The first graph contains the SS of the crushed stone, a yellow wall, and a red wall ([Fig. 2a](#)). The crushed stone and the red wall had a very similar SS. Therefore, we can infer that they are very similar color as well as materials. Furthermore, the yellow wall has three different peaks at 570, 605, and 650 nm, corresponding with the yellow color in the approximated color chart, located at the left-top of the graph. Moreover, the yellow wall has more energy at 440 and 890 nm. This energy at 440 nm produces a dark-yellow color (similar to mustard) in the visible light spectrum. The energy sensed at 890 nm may indicate that this color is more susceptible to absorb heat.

The second graph contains the SS of a yellow flower, a white flower, and a green plant ([Fig. 23b](#)). The yellow flower has three peaks at 570, 600, and 645 nm. This pattern will produce a color similar to the yellow and orange color in the visible light spectrum. Furthermore, the white flower has much energy in the visual spectrum of light as expected. Nevertheless, the more significant peak it has is located at the 450 nm (blue). This is a remarkable result because the flower has a purple line in its leaves. These lines are not evident in the RGB image, and we did not observe it during the acquisition process. Therefore,

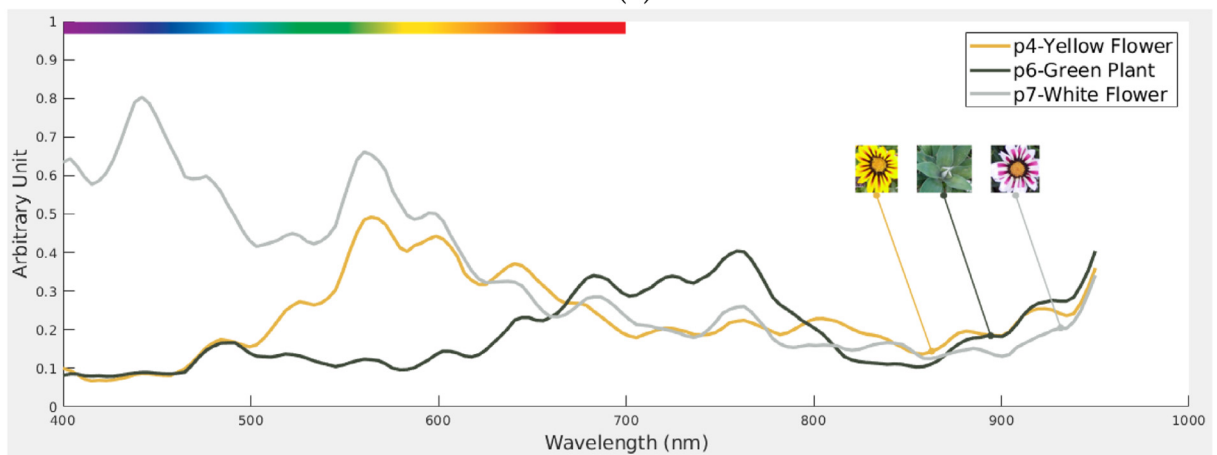
⁵ The spatial-resolution is the number of pixels in an image.

⁶ Image stitching is a technique to create a high-resolution image by blending multiple photographs with overlapping fields of view.

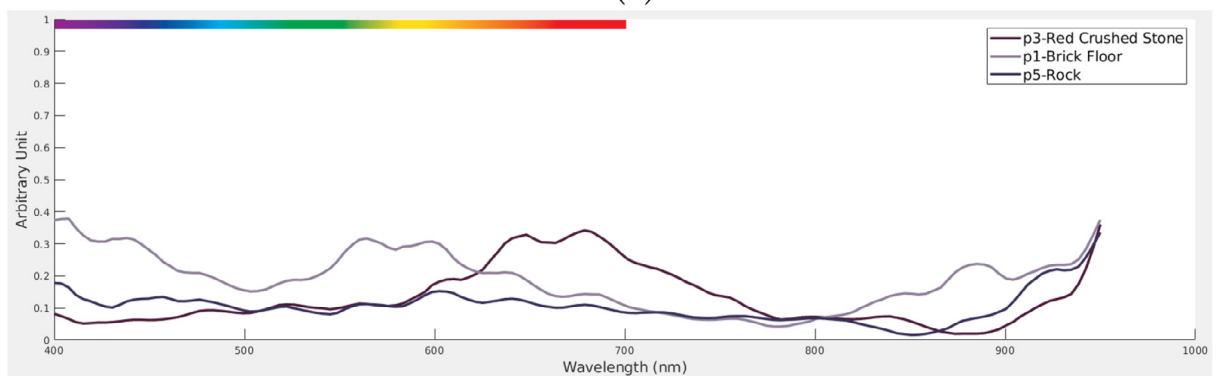
⁷ Spectral signature is the variation of light as function of wavelength.



(a)



(b)



(c)

Fig. 23. It shows three graphs, one for each subset of objects in the sensed landscape. (a) Walls: Red crushed stone, yellow wall, and red wall. (b) Plants: Yellow flower, green plant, and white flower. (c) Stones: Red crushed stone, brick floor, and a Rock. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the proposed HSI device has shown a significant performance. Moreover, the SS of the green plant looks different from the expected. Nevertheless, it may be explained by the photosynthesis process, which reflects the near-infrared wavelengths (longer than 680 nm).

Finally, the third graph has the SS of crushed stone, a brick floor, and a rock. The brick floor has four peaks at 415, 560, 600, and 880 nm. Therefore, the visible color is a combination of the purple and red which is consistent with the expected RGB

color. Furthermore, the stone has a SS with low energy. In consequence, it has a dark visible color as expected in the RGB image.

10. Conclusions

Consumer equipment, commercial 3D printers, and open-source software have been successfully used to design and develop a single shot HSI system which operates from 400 nm to 1052 nm. Its highest spectral performance has been observed from 415 nm to 920 nm. This device can extract up to 315 wavebands to measure transmittance. For reflectance applications, it obtained better results when the hyperspectral image contains 64–105 evenly separated wavebands. With only 2% of the cost of commercial devices with similar characteristics, the proposed device showed high spectral accuracy in controlled light conditions as well as ambient light conditions. Moreover, this price could be reduced by an additional 60% if the user replaces the recommended front lens with a cheaper one. Furthermore, it has a spatial resolution that is enough for many applications. Nevertheless, the image quality was influenced by the texture in the diffraction grating, and it is an opportunity for future works. Furthermore, unlike related works, the proposed HSI system includes a framework to build the proposed HSI device from scratch. This framework decreased the complexity of developing it as well as the processing time. It also contains every needed 3D model, a calibration method, the image acquisition software, and the methodology to build and calibrate the proposed HSI device. Besides, the proposed HSI device weights up to 300 g. Making it highly portable, highly reusable, and of lightweight for possible research and commercial applications.

Human and Animal Rights

None.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

1. This paper acknowledges to the Mexican Council of Science and Technology (CONACYT), for the scholarship provided during the research.
2. This paper acknowledges to EdmundoOptics.com, for the Silver Medal granted in the EdmundoOptic Educational Awards 2016.

References

- [1] Gamal ElMasry, Ning Wang, Adel ElSayed, Michael Ngadi, Hyperspectral imaging for nondestructive determination of some quality attributes for strawberry, *J. Food Eng.* 81 (1) (2007) 98–107.
- [2] Hiroshi Okamoto, Won Suk Lee, Green citrus detection using hyperspectral imaging, *Comput. Electron. Agric.* 66 (2) (2009) 201–208.
- [3] Yunseop Kim, David M. Glenn, Johnny Park, Henry K. Ngugi, Brian L. Lehman, Hyperspectral image analysis for water stress detection of apple trees, *Comput. Electron. Agric.* 77 (2) (2011) 155–160.
- [4] C.-I. Chang, *Hyperspectral Data Processing: Algorithm Design and Analysis*, John Wiley & Sons..
- [5] J.M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, J. Chanussot, Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches, *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 5 (2) (2012) 354–379.
- [6] Ainara López-Maestresalas, Janos C. Keresztes, Mohammad Goodarzi, Silvia Arazuri, Carmen Jarén, Wouter Saeys, Non-destructive detection of blackspot in potatoes by vis-nir and swir hyperspectral imaging, *Food Control* 70 (2016) 229–241.
- [7] Fred Sigernes, Mikko Syrjäsuo, Rune Stovold, Jo ao Fortuna, Mariusz Eivind Grøtte, Tor Arne Johansen, Do it yourself hyperspectral imager for handheld to airborne operations, *Opt. Express* 26 (5) (2018) 6021–6035.
- [8] K. Uto, H. Seki, G. Saito, Y. Kosugi, T. Komatsu, Development of a low-cost hyperspectral whiskbroom imager using an optical fiber bundle, a swing mirror, and compact spectrometers, *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 9 (9) (2016) 3909–3925.
- [9] K. Uto, H. Seki, G. Saito, Y. Kosugi, T. Komatsu, Development of a low-cost, lightweight hyperspectral imaging system based on a polygon mirror and compact spectrometers, *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* 9 (2) (2015) 861–875.
- [10] Ralf Habel, Michael Kudenov, Michael Wimmer, Practical spectral photography. *Computer Graphics Forum (Proceedings EUROGRAPHICS 2012)*, 31(2) (2012) 449–458..
- [11] Amr Abd-Elrahman, Pand-Chhetri Roshan, Gary Vallad, Design and development of a multi-purpose low-cost hyperspectral imaging system, *Remote Sens.* 3 (2011) 12.
- [12] Liang Gao, Robert T. Kester, Nathan Hagen, Tomasz S. Tkaczyk, Snapshot image mapping spectrometer (ims) with high sampling density for hyperspectral microscopy, *Opt. Express* 18 (14) (2010) 14330–14344.
- [13] Scott A. Mathews, Design and fabrication of a low-cost, multispectral imaging system, *Appl. Opt.* 47 (28) (2008) F71–F76.
- [14] William Johnson, Daniel W. Wilson, Wolfgang Fink, Mark Ph.D, Greg Bearman, Snapshot hyperspectral imaging in ophthalmology, *J. Biomed. Opt.* 12 (2007) 014036..
- [15] AppImage, A way for upstream developers to provide "native" binaries for linux. <https://appimage.org>, 2019..
- [16] Gamal ElMasry, Ning Wang, Adel ElSayed, Michael Ngadi, Hyperspectral imaging for nondestructive determination of some quality attributes for strawberry, *J. Food Eng.* 81 (1) (2007) 98–107.
- [17] A. Kuehn, A. Graf, U. Wenzel, S. Princz, H. Mantz, M. Hessling, Development of a highly sensitive spectral camera for cartilage monitoring using fluorescence spectroscopy, *J. Sens. Sens. Syst.* 4 (2) (2015) 289–294.

- [18] GNU, Octave:scientific programming language.<https://www.gnu.org/software/octave/>, 2019..
- [19] MathWorks, Matlab.<https://www.mathworks.com>, 2019..
- [20] Takayuki Okamoto, Akinori Takahashi, Ichirou Yamaguchi, Simultaneous acquisition of spectral and spatial intensity distribution, *Appl. Spectrosc.* 47 (8) (1993) 1198–1202.