Dokument prokazující dosažení výsledku

Knihovna software pro zpracování barevného obrazu

Technická zpráva M1_TH03010330-V4

Martin Kolář, Pavel Zemčík, Jan Schier, Peter Honec

Vysoké Učení Technické v Brně

Brno 2021

T A
Č R

Program **Centra kompetence**

Tento dokument byl vytvořen s finanční podporou TA ČR v rámci výzkumného program TH03010330 (BOREC – Barevný Obraz v prostředí „Realtime Embedded Computing“).

Číslo a název projektu:

TH03010330	BOREC – Barevný Obraz v prostředí „Realtime Embedded Computing“

Název a popis dílčího výstupu:

Knihovna software pro zpracování barevného obrazu
Knihovna bude zahrnovat obecně použitelné funkce pro zpracování barevného a HDR obrazu pro nasazení v aplikacích. Bude se jednat zejména o funkce pro extrakci barevných příznaků, detekci objektů a předzpracování a redukci barevného prostoru/tone mapping. Knihovna se bude dále využívat v rámci projektu, ale bude i nabízení třetím osobám jako open source. Hardwarový výstup projektu a softwarové výstupy mají úzkou vzájemnou vazbu. Proto je text v těchto položkách u jednotlivých výstupů totožný.

Jazyk dokumentu

Angličtina

Organizace a řešitel

Vysoké Učení Technické v Brně	Prof. Dr. Ing. Pavel Zemčík

1 Introduction

In this report, we describe the library created to process high-dimensional images, as captured by hyperspectral cameras, HDR cameras, and other custom hardware. The library consists of five packages: the interactive visualisation package, the HDR processing package, the deep spectral classifier package, the sparse spectral segmentation package, and the spectral timelapse viewer package.

The motivation and applications described in this document are followed by analysis of existing partial solutions. Hyperspectral cameras and software is listed, with their shortcomings and merits listed. Then, a list of light sources and sensors is catalogued and calibrated, leading to all preparation necessary for the creation of a functional sample demonstrating the proposed workflow.

This report assumes prior knowledge of the spectral decomposition of light, color perception, and digital image capture.

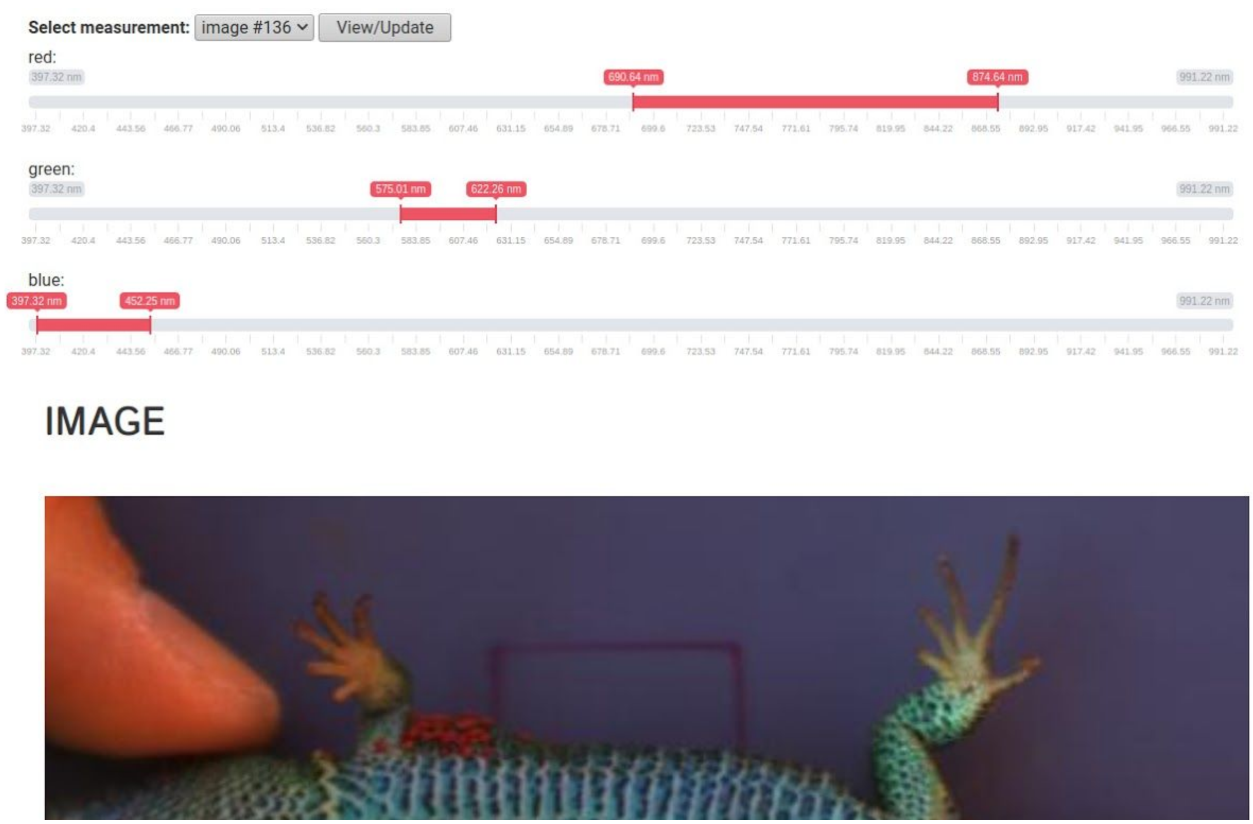
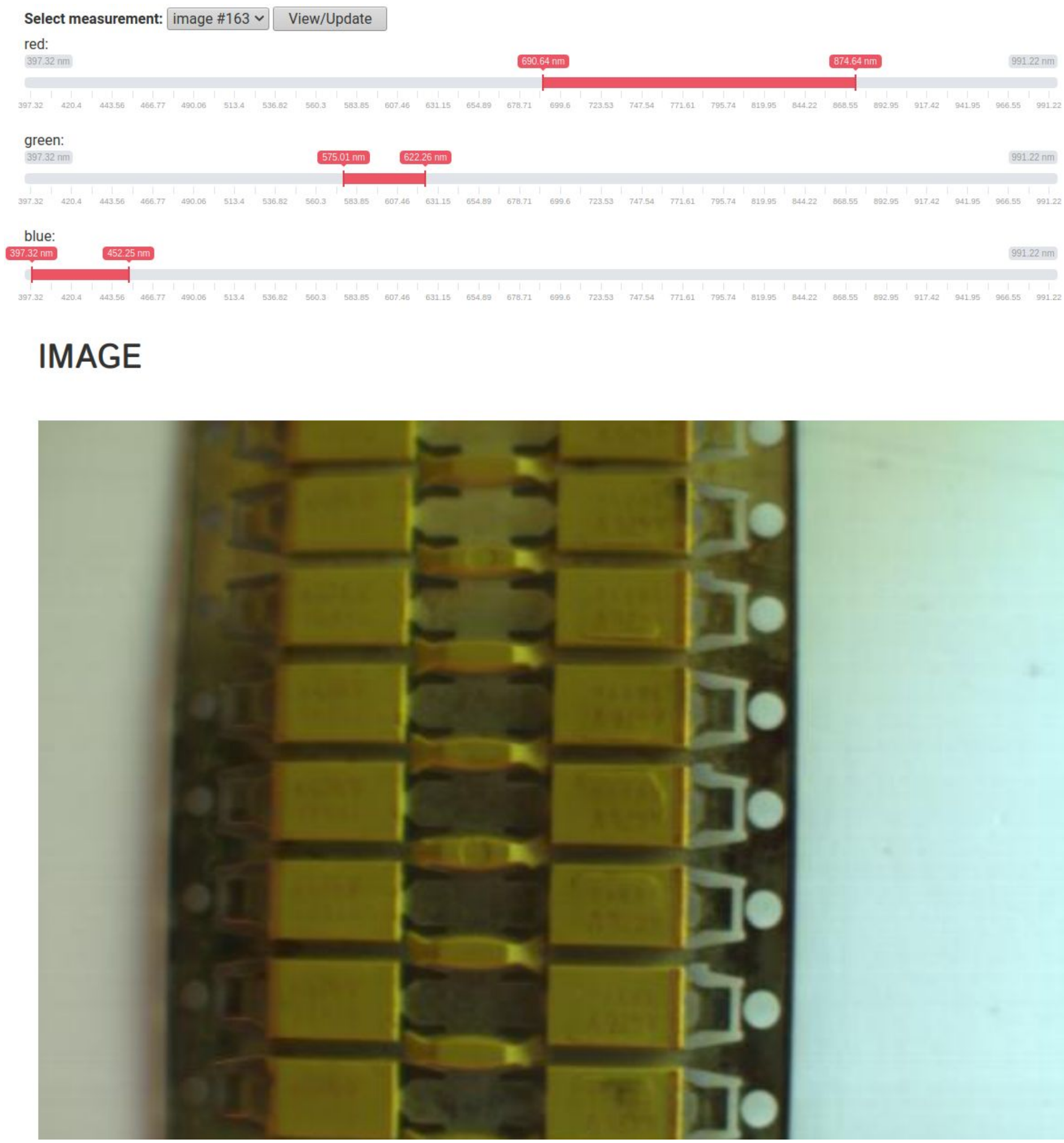


Figure 1.: An application of the hyperspectral interactive visualisation package

2 Motivation

Capturing, processing, and analyzing hyperspectral and HDR images is challenging, because classical solutions are targeted at three-channel visual data: Red, Green, and Blue. In order to extract meaningful actionable information from higher-dimensional images, it is necessary to develop custom solutions.

Consider, for example, the problem of detecting defects in SMD components. Automatic detection requires an imaging system which differentiates undesirable cracks from printed text. These are indistinguishable in RGB captures under visible daylight.



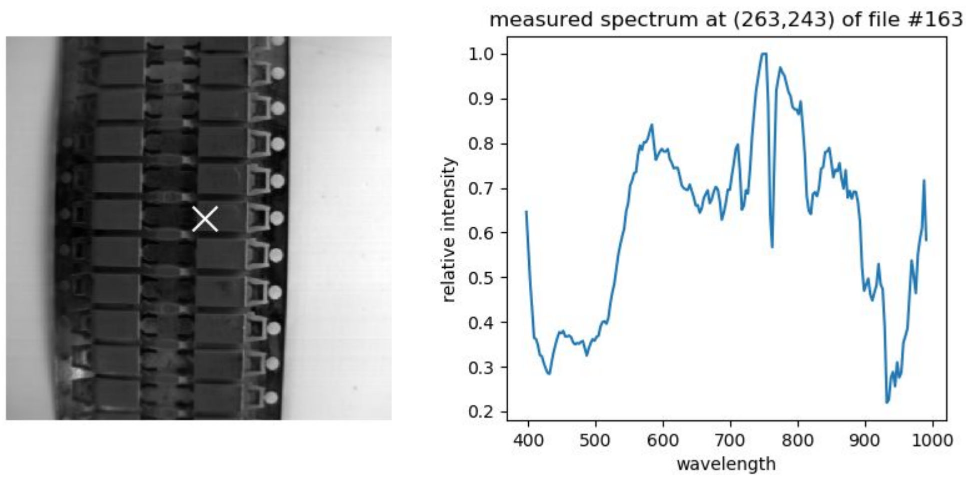
3 Specifications

This package consists of five interconnected packages, written in Python. The specifications of each of these is described below:

3.1 The Interactive Visualisation Package

This package is a Flask container serving a webpage generated in Python. This webpage offers two core functionalities: RGB visualisation of a multispectral image, and spectral response graphing at any given pixel. Both operations are graphically intuitive, by moving sliders, or clicking on the image.

Sliders to define how a multispectral image is converted to RGB

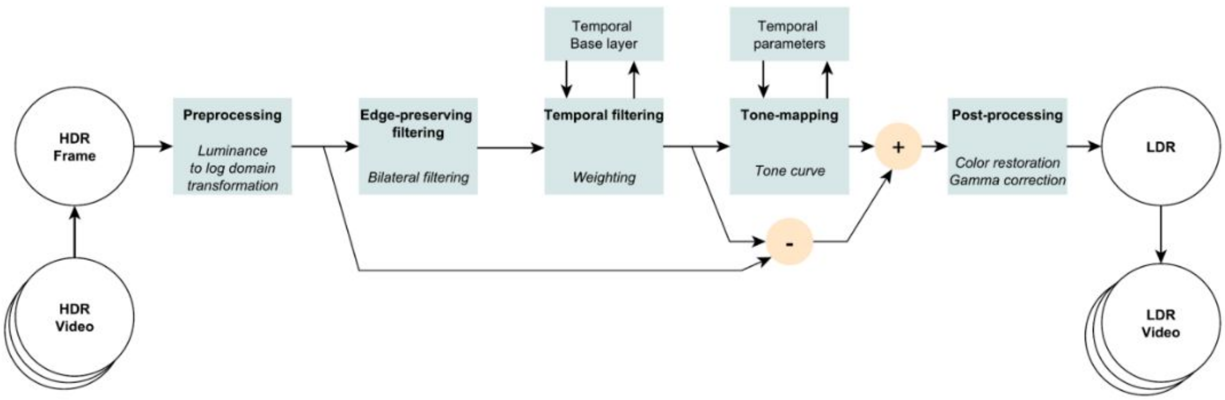


Clicking anywhere on the image produces a graph of spectral response at the selected location

Furthermore, any given settings (image ID, RGB composition, x-y position) are stored in the URL, allowing the sharing of specific static visualisations by simply sending the link.

3.2 The HDR Processing Package

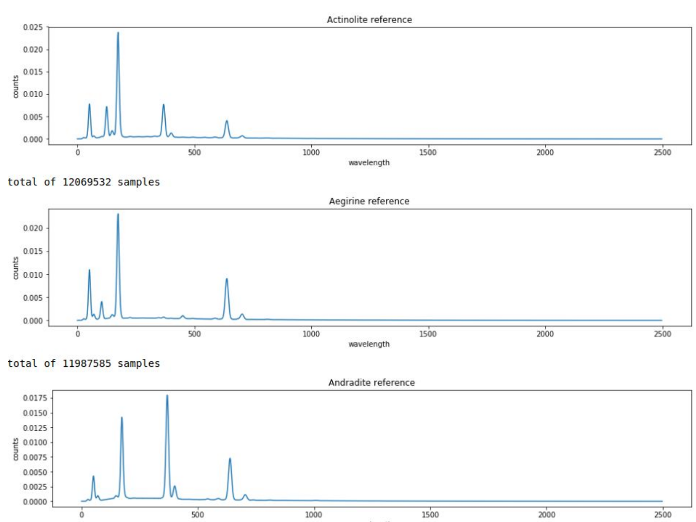
This package is used for high dynamic range (HDR) tone / video mapping. The application implements an algorithm based on Durand's local operator extended by video processing support. For the purpose of displaying HDR images on a standard display device, postprocessing is implemented in the application to improve the contrast. The output can be exported to a bitmap or compressed video.



The architecture of the application for tonemapping HDR images and video

3.3 Deep Spectral Classifier Package

This package is designed to differentiate and cluster individual spectra. The data can correspond to spectroscopic point measurements, hyperspectral image data, or X-Ray spectroscopic measurements.

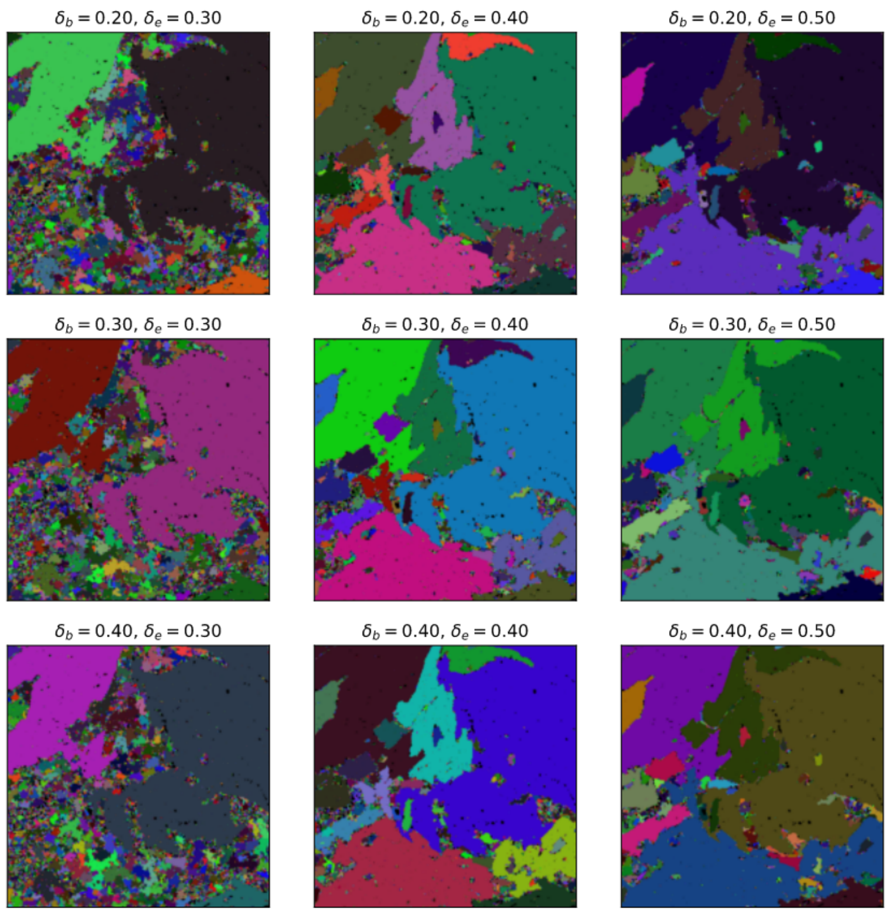


Sample spectra being classified

3.4 Sparse Spectral Segmentation Package

This package is designed to perform segmentation of image regions whose spectra differ, without performing classification. The reason for performing these operations separately is the unsupervised setting in which it may be used: it may be needed to process unseen spectra and unseen spectral differences.

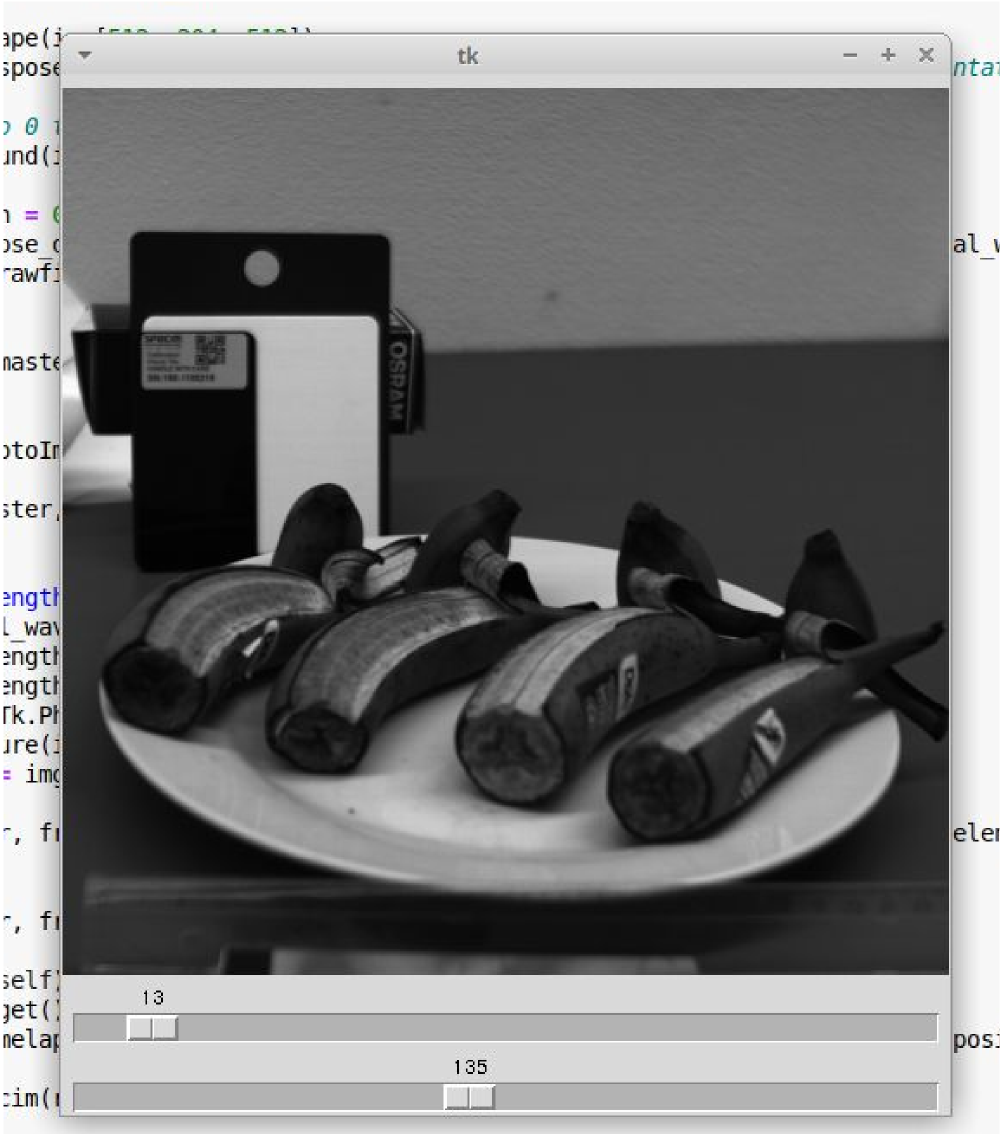
The method uses neural networks to perform local difference predictions, with Markov Random Field grid optimisation to separate segments. The expected modality is a single grayscale image, combined with sparse spectral measurements.



Segmentations of the same data with different parameters for granularity and specificity

3.4 Sparse Spectral Segmentation Package

This package enables the displaying of sequences of captures with a multispectral or hyperspectral camera. The benefit of this work is to enable the study of how spectra change over time for some natural and industrial processes, such as moulding, annealing, or growth.

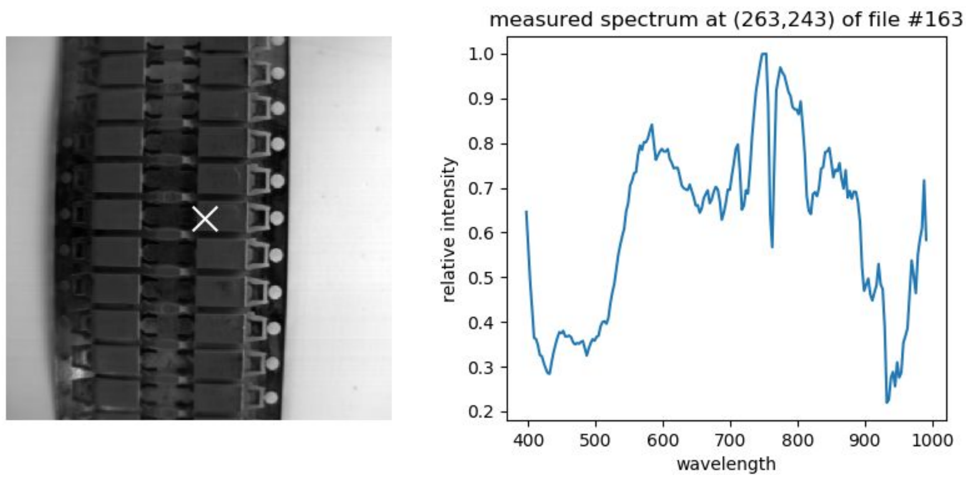


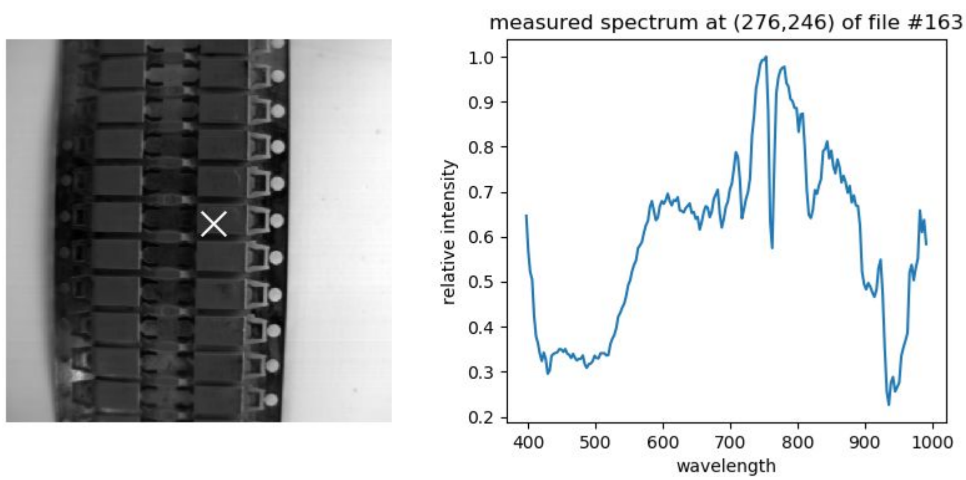
Multi-hour sequence being displayed with interactive sliders for spectrum(top) and time(bottom)

4 Usage

5 Validation

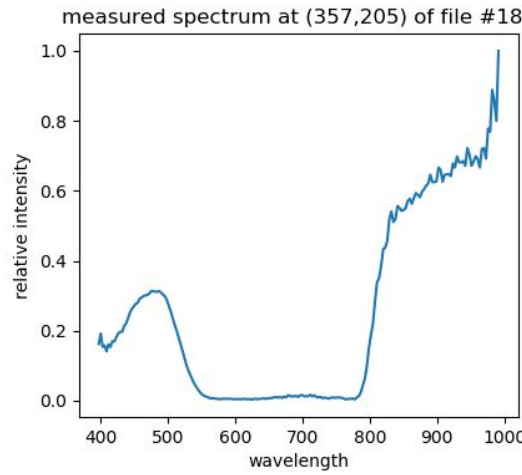
The following results demonstrate a practical commercial application of the library.





Thanks to the simple and intuitive interface, it is easy to capture a scene, and quickly identify relevant differences between desired spectra. In the above example, SMD components are processed, visualised, and two wavelengths are identified: 600nm where the text sharply differs from the background, and 850nm where they are indistinguishable.

This allows the creation of lighting conditions where the desired artefact can be highlighted (600nm) and ignored (850nm). By retrieving these wavelengths from the database created for this project in the past, the relevant LED lights can be used, and combined with an arbitrary sensor.



The online user interface allows the capture of specific scenes and markers from hyperspectral data, and their intuitive display and sharing across online users through a simple URL address[1]

7 Availability

The library and its packages are available at the following locations:

1. the interactive visualisation package - https://borec.fit.vutbr.cz/banany - https://github.com/mrmartin/spectral_online_viewer
2. HDR processing package - wis.fit.vutbr.cz/FIT/db/vav/view_product.php?id=652
3. deep spectral classifier package - https://github.com/mrmartin/spectral-class
4. sparse spectral segmentation package - https://github.com/RomanJuranek/graph-segmentation
5. spectral timelapse viewer package - https://github.com/mrmartin/spectral_timelapse_viewer

~~8 Analysis~~

~~Given the selected algorithm, the spectral analysis library considers combinations of catalogued lights and sensors to optimize differentiability. A virtual scene will be generated for any combination of lights and sensors, simulated with the captured hyperspectral data. The classification/segmentation will be evaluated on the virtual scene, allowing optimization over scene parameters.~~

[1] https://borec.fit.vutbr.cz/banany/point_spectrum/filenum=181&x=357&y=205

9 Conclusion

This technical report has presented the motivation for a new hyperspectral workflow, to enable "Realtime Embedded Computing" for the use of spectral information in industrial applications. Existing approaches, hardware, and software have been analysed and compared.

The result of this work is a proposed workflow, which captures the scene with a hyperspectral camera, analyses it with specialised software, and designs an embedded capture system which makes use of the spectral information.

Further steps have been taken to enable the creation of a functional sample of this workflow. A set of light sources has been selected and catalogued, and the spectral sensitivity of a selection of sensors has been calibrated.