

# POLITECNICO DI TORINO

## Master of science in Mechatronic Engineering



Master Thesis Project

## Image Analytics for In-situ Defect Detection in Additive Manufacturing

### Supervisors

Prof. Edoardo Patti  
Prof. Santa Di Cataldo  
Prof. Massimo Poncino

### Candidate

Giovanni Ferrari

Academic Year 2018-2019



*“There is nothing noble in being superior  
to your fellow man: true nobility is being  
superior to your former self”*

- Ernest Hemingway



# Index

I. Abstract .....	I
II. Acknowledgements .....	II
III. Figures index .....	III
IV. Tables index .....	V
1. Introduction .....	1
2. State of the art .....	2
2.1. Additive Manufacturing .....	2
2.2. Image analysis .....	8
2.3. Machine Learning for Additive Manufacturing .....	11
3. Process description .....	12
3.1. Techniques differentiation .....	13
3.2. Defect analysis in Selective Laser Melting .....	19
4. Infrastructure description .....	20
4.1. EOS M290 machine .....	20
4.2. UI-1540SE-M-GL camera .....	24
5. Acquisition script .....	25
5.1. Raspberry Pi version .....	25
5.2. PC version .....	31
6. Acquisition hardware setup .....	33
6.1. Raspberry Pi version .....	33
6.2. PC version .....	38
7. Image analysis .....	40
7.1. Filters .....	40
7.2. Defects identification .....	43
7.3. Feature extraction .....	46
7.4. Validation .....	53
8. Conclusions and further development .....	54
9. References .....	57



# I. Abstract

In recent years the affirmation of industry 4.0 has marked a trending towards new means of production and more sophisticated control infrastructures. An example of this trend is the widespread application of additive manufacturing machine, alongside innovative software and control techniques, as testified by the application of Machine Learning in production processes and the shift towards an IoT approach.

This thesis project is part of the application of a Machine Learning algorithm to a Selective Laser Melting (additive manufacturing technique) machine in order to optimize the machine parameter to prevent defect formation. In particular, the part covered by this thesis is the feature extraction from the process: using a photo camera, pictures of the building process in a Selective Laser Melting machine are taken, then such pictures are analyzed through Python methods to extract relevant features that will be later fed to the optimization and control part.

## I. Acknowledgments

I would first like to thank my thesis supervisors prof. Edoardo Patti and prof. Santa Di Cataldo of Politecnico di Torino for the time they spent following me during this project, as well as Davide Cannizzaro, who helped me in the times I was stuck during the thesis. All of them always gave me the right direction to follow, keeping me focused on the development of the thesis whenever I needed them.

I must thanks, of course, Stefano Paradiso of FCA Italy SpA. He was the one that more than anybody allowed this project to proceed by giving me advices for each important decision in the project, and following my work, day by day, monitoring the developments and helping me with every practical issue that came up.

I'm extremely thankful that I had my friends on my side, the one I left home to pursue the highest education in this University, that never abandoned me, and the friends I encountered here, the best companions I could ever hoped for to share these circumstances. You filled my time with laughter and joy.

Finally, my deepest gratitude goes to my family, my parents and my brother, and Maria Chiara, that allowed me to complete my studies supporting me in a practical and emotional way. Every time, during the time spent in the university, that I felt doubtful or discouraged, I relied on them to remember me that whatever difficulty I was facing, the people I love would always be there to shelter me and renew my determination. This achievement is yours as much as it is mine, and I hope I'll be able to repay you from this day on.

Giovanni Ferrari

## II. Figure index

2.1. Spattering generation; .....	2
2.2 Spattering tracking through visual monitoring of the process; .....	3
2.3. Spherical porosities; .....	4
2.4. Two cases of Lack of Fusion; .....	4
2.5. Two images of crack formation; .....	5
2.6. Snake contouring; .....	8
2.7. Example of layer-by-layer monitoring; .....	10
3.1. Traditional steps for Additive Manufacturing production; .....	12
3.2. Possible designs for supports; .....	13
3.3. Tradeoff between configurations of supports; .....	14
3.4. Two implementations of stereolithography; .....	14
3.5. Fused Deposit Modeling schematic; .....	15
3.6. Necking between two grains in Selective Laser Sintering; .....	16
3.7. Selective Laser Melting schematic; .....	17
3.8. Electron Beam Modeling schematic; .....	18
4.1. EOS M290 datasheet; .....	20
4.2. SLM machine building site; .....	22
4.3. Schematic of the gas flow implementation; .....	22
4.4. IDS camera datasheet and appearance; .....	24
4.5. Camera software's interface; .....	24
5.1. Conversion folder organization and flow chart of the code; .....	30
5.2. Difference between bad timed acquisitions and correctly timed acquisitions; .....	26
5.3. Final acquisition code structure in Raspberry Pi version; .....	27
5.4. Flow chart of the first operational code; .....	28
5.5. Flow chart of the final code; .....	28
5.6. LOG example; .....	29
5.7. Visual proof of loss of synchrony in latest layers; .....	29
5.8. Final acquisition code structure in PC version; .....	31
5.9. Icon of the desktop program; .....	32
5.10. Example of shebang; .....	32
5.11. Environmental condition on the PC used for acquisition; .....	32

## Image analytics for in-situ defect detection in Additive Manufacturing

6.1. Recoater's driver appearance; .....	33
6.2. Instrumentational amplifier setup; .....	33
6.3. MOSFET setup; .....	34
6.4. Photoresistor setup; .....	34
6.5. Photoresistor design manufactured with FDM technology; .....	35
6.6. Initial hardware setup for the Raspberry Pi version; .....	35
6.7. Hardware setup for Raspberry Pi version with MOSFET addition; .....	36
6.8. Final hardware setup for Raspberry Pi, schematic and illustration; .....	37
6.9. General overview of the Raspberry Pi hardware setup; .....	37
6.10. Arduino setup schematic and illustration; .....	38
6.11. Cable installation and electrical panel; .....	39
6.12. General overview of the PC setup; .....	39
7.1. Sobel operator; .....	40
7.2. Label function; .....	41
7.3. Contouring function; .....	41
7.4. Snake function; .....	42
7.5. Example of incandescence; .....	43
7.6. Example of holes; .....	44
7.7. Example of spattering; .....	44
7.8. Example of horizontal defects; .....	45
7.9. Example of waves; .....	45
7.10. Features of interest; .....	46
7.11. Image analysis code structure; .....	46
7.12. Normalization effect on different acquisitions; .....	48
7.13. Active contouring application; .....	49
7.14. Holes identification; .....	50
7.15. Incandescence identification; .....	50
7.16. Spattering identification; .....	51
7.17. Horizontal identification; .....	51
7.18. Waves identification; .....	52
8.1. Visual identification of defects' features; .....	54
8.2 Interference in the identification between defects; .....	55

### III. Table index

- 5.1. LEDs signaling in the Raspberry Pi acquisition system; ..... 27
  
- 6.1. Hardware components of the Raspberry Pi acquisition system; ..... 37
- 6.2. Hardware components of the PC acquisition system; ..... 38
  
- 7.1. Number of defects in validation pool; ..... 53
- 7.2. Table used for validation of the image analysis algorithm; ..... 53
  
- 8.1. Accuracy results of the image analysis algorithm; ..... 55
- 8.2. Time performances of the image analysis algorithm. .... 56



# 1. Introduction

The Additive Manufacturing technology has been a breakthrough in the industrial sector, born for rapid prototyping and now starting to be implemented for production of pieces with a new set of rules for design made possible by the many advantage of the technology. The passage to the mainstream diffusion is still recent for Additive Manufacturing (AM), generating a growing interest in the applications surrounding the AM paradigm, from the characterization of new powder, to the application of innovative programs for control and optimization of the process.

The latter is precisely the origin of this thesis project.

Contemporary to the affirmation of new manufacturing technologies, new data analysis techniques are becoming more popular and increasingly more effective, like machine learning and its component (data mining, neural networks...). A growing amount of literature is now dedicated to the application of machine learning techniques to extract relevant features from production technologies and the usage of these features for early defect recognition, predictive maintenance, and parameters optimization., investigating the hidden link between measurable characteristics of the process and the overall performances.

This project aim at performing a defect detection in the an AM process, namely a Selective Laser Melting process, by mining data in the form of images during the whole process, data that will be used to train an algorithm for early fault detection, allowing an early intervention, and possibly the design of a controller that exploit said data.

The thesis is presented following the chronological development of the project, divided in two main parts. At first a study of the state of the art regarding AM technologies and how machine learning has been used in the field is presented, comprehending also some interesting publications on image analysis. The AM processes, in particular, will be looked at in depth to understand the context of the project, and the same will be done for the machinery used.

The first practical part of the thesis is the design of an acquisition system that allows to collect raw sources of information from the manufacturing process, hence a camera will be used to capture images of the process in suitable moments, and store the images as the base of the data; this part will be analyzed both from the hardware design point of view and the software design, explaining the choices made based on requirements and constrains given by the environmental condition or by sensitivity of other elements of the overall system.

The second part is the extraction from the image pool of the relevant features, that will be described. The filters exploited will be described, and based on those, the extraction algorithm will be presented, justifying the choices made for the design and the limitation that needed specific corrections. The main goal of this part is to create a valid algorithm with a design approach that favors further developments and optimization of the algorithm itself.

Finally, the performances and the results of the designed algorithm will be presented and some preferential directions for future improvements will be proposed, based on the information gained during the creation and the testing of the program.

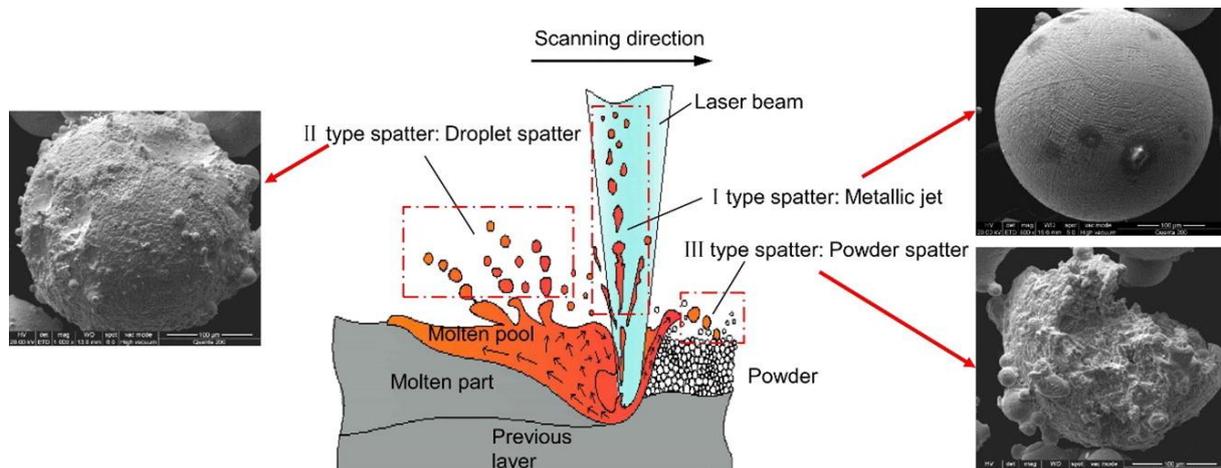
## 2. State of the art

In the following a brief review of the state of the art is performed on topics interesting to the purpose of the project. In particular three areas will be covered: the first section covers papers regarding defect formation in Additive Manufacturing process, to provide some context on the importance of process parameters. In the second section some image analysis functions that provided useful tools are explained. Finally, some examples of how Machine Learning algorithm are used in Additive Manufacturing are analyzed, again to provide context and a base for future development. Although the papers reviewed in this chapter were used in the thesis, a lot of study material also came from publications such as books and manuals, which are not covered here being already well established as working procedures.

### Additive Manufacturing:

*D. Wang – Mechanism and characteristics of spatter generation in SLM processing and its effect on the properties [1]*

The laser power and the spattering formations are put in correlation, establishing a positive correlation between the power given to the pool and liquid metal jetted out. Responsible for the generation of spattering are three factors: recoil pressure due to the high input energy, Marangoni effect due to gradient in surface tension between particles, which result in a kinetic manifestation, and heat effect in molten pool. The observed size of the spattering is relatively big in a powder bed, reaching  $162\mu\text{m}$ , and assuming a different composition with respect to the original powder. This behavior causes a deterioration of the final fabricated part.

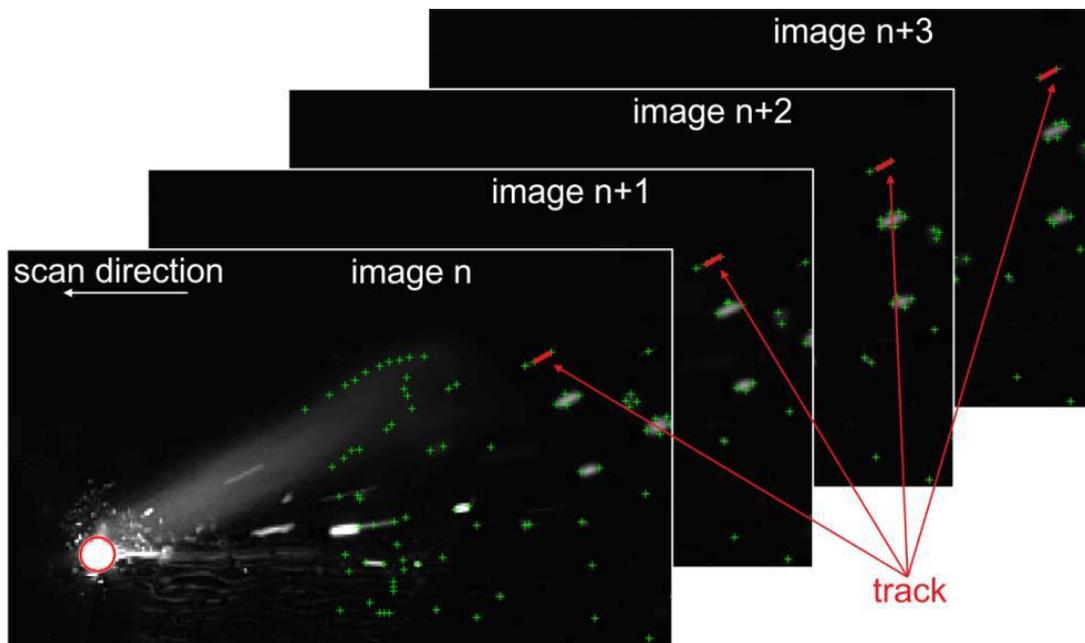


(fig.2.1) Spattering generation is caused by ejection from the melting pool. Wang et al. [1]

*T. Heeling – Investigation of selective laser melting spatter characteristics for single- and multi-beam strategies using high speed imaging [2]*

An overview of the process dynamic using high speed camera is given, focusing on the spattering formation and characteristics. A Phantom V12 camera recording 30010 fps with 512x256 frame and a 15 $\mu$ m/pixel resolution is used. Different scan strategies are used, such as single-beam scan, multi-beam with post-heating, pre-heating and wobbling.

The acquisition are preprocessed by adjusting the greyscale, boosting the value of darker particles and setting to zero the samples below a threshold for noise reduction, and a filter is also applied to exclude the area near the laser from identification, as this will lead to the identification by gradient of the laser in action and the plume not related to spattering. To further reduce the noise and identify the single droplets, artifacts are tracked by computing the distance frame by frame and associating the object with a near artifact in the previous (fig. 2.2). This allows to track droplet of melted material mid-air and relate them to where the laser was when they were originated: the average identification rate is about 75%.



(fig. 2.2) the spatter is tracked mid-air. It's possible to notice how the area around the laser is excluded and dark spatter is enhanced on the right. Heeling at al. [2]

Two kind of spatter are tracked, bright and dark spatter, depending on dimension and speed (and thus residual incandescence): results shows how scan speed has a positive correlation with the spatter count, although for higher scan speed there's a lower spatter velocity. This is due to the reduction of energy density at higher speeds, which result in smaller and less deep melt pool, thus reducing the cohesion with layers underneath. This causes the increase of chances for liquid metal to escape the melting pool, being less homogeneous.

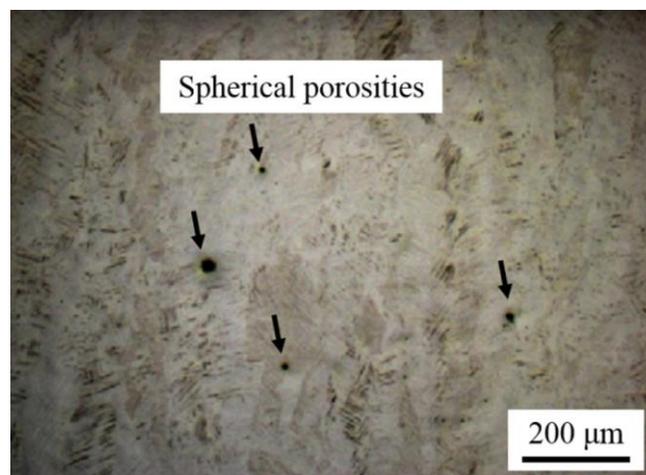
Finally, a comparison with multi-beam strategies is performed, showing how preheating reduces the spattering by partially agglomerate the particles, while post-heating increases the spattering count by disintegrating larger particles into smaller ones. Wobbling strategies can be exploited to enlarge the positive effect of preheating.

*B. Zhang - Defect formation mechanisms in selective laser melting: a review [3]*

The paper reviews the known defect in selective laser melting, proposes a classification in three groups and a comprehensive defect detection and control approach. The focus of the paper is on the effects of incorrectly tuned parameters that affect the energy density (described later), namely of the laser device.

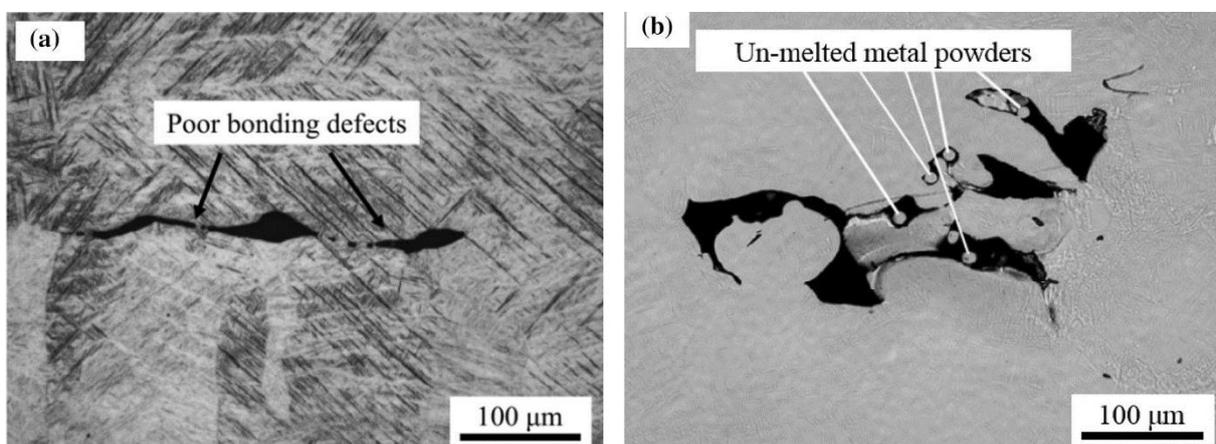
Three classes of defects are described:

- Porosities: a porosity is a gas formation trapped inside the artifact, due to high energy density (low scan speed and high laser power). The high energy density may cause the vaporization of low melting point component of the alloy, which are unable to reach the surface due to the extreme cooling rate. Another cause of porosity is the low density of the powder in the deposition phase, causing the gas present between the grains to remain trapped as described before;



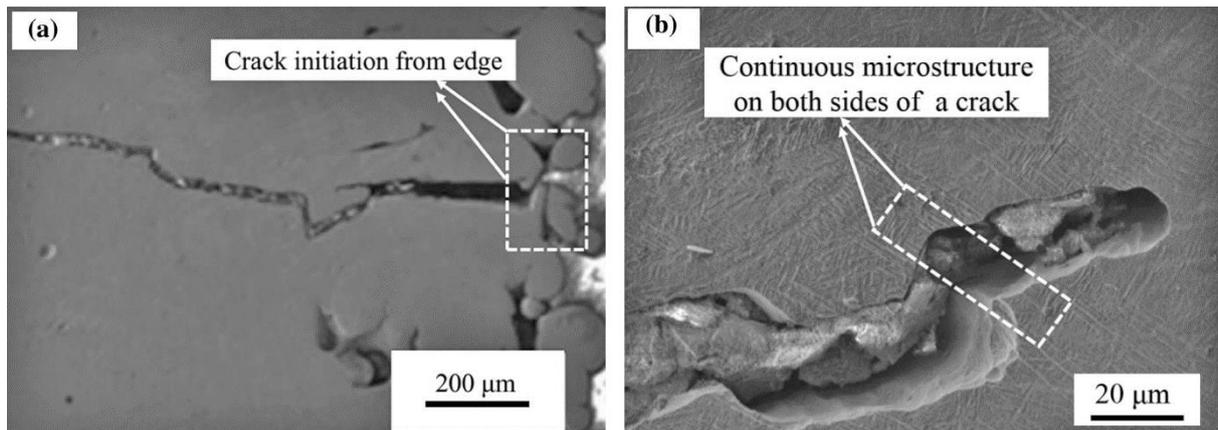
(fig. 2.3) spherical porosities in an optical image. Zhang et al [3]

- Incomplete fusion holes (fig. 2.4): also called lack of fusion (LOF), this defect is caused by low energy density that results, as the name suggests, in a lack of fusion in some areas, leaving un-melted metal powder or poor bonding contact. Particularly sensitive to this defect are the border between different laser tracks and layers;



(fig. 2.4) two cases of LOF. Zhang et al. [3]

- Cracks (fig. 2.5): this defect is caused by the extreme thermodynamic conditions of the melting pool, causing artifacts to build up severe residual thermal stress as a result. Although this may affect every artifact, depending on chamber conditions (low chamber temperature can result in this phenomena), alloys with high thermal expansion coefficient and low thermal conductivity are more prone to it. Crack likely start from built surface in contact with the partially melted metal powder, meaning it can be difficult to spot from the surface.



(fig. 2.5) two images of crack formation, initiation from the edge and a focus on the microstructure. Zhang et al. [3]

A brief description of energy density is then given, as the ration between laser power and the product of scan speed, hatch spacing, and layer thickness. This value is described as particularly useful to describe the process parameters, since it accounts the energy given to a volume unity, directly responsible for the melting conditions.

Other than the parameters, powder materials composition is accounted for sensitivity to defect insurgence, as well as scan strategy. The latter, in particular, are intended as the kind of path the laser should follow layer by layer. The original observation is the higher chance of presence of defects in the borders between tracks in the same layer and between layers itself. Repeated scan strategy, so, are more likely to produce widespread defects in the artifact. Suggested paths are the partition of the surface in “island” and crossing of path in subsequent layers. Finally, a link between defects and mechanical properties is described (both tensile and fatigue properties), and a strategy is proposed based on the incorporation of numerical models, online monitoring, and the control of process parameters to promptly detect defects formation and implement control strategies and parameters optimization.

*G. Dongdong – Balling phenomena in direct laser sintering of stainless-steel powder: Metallurgical mechanism and control methods [4]*

An investigation on the effects of process parameters and material quality on the balling phenomena is conducted, providing two kinds of powder with different sizes of the grains and a small quantity of de-oxidant in the finest powder. A uniform layer of 0.25mm is used, and different single line scan tests 50mm long are manufactured, with different combinations of scan speed and laser power. Subsequently other experiments are performed tuning the layer thickness. A digital camera is then used to record the result and a scanning electron microscope afterwards to characterize the artifact, after chemical preparation. Samples are categorized in four groups, divided by quality of the track evaluated with scanning electronic microscope. It is observed how for a given scan speed, the lowering of laser power provoke the increase of viscosity of the melt pool, causing the aggregation in spheres the diameter of the laser beam (first kind of balling phenomena, fig. 2.6) and preventing efficient bonding, possibly resulting in porosity, distortion and lamination.

The second kind of balling phenomena occurs when high scan speed is used: this doesn't allow stability of the melt pool, due to lower energy density and the influence of scan speed on the capillary stability of the liquid. The result of this instability is the formation of metallic droplet to escape the pool and preserve the shape after solidification. It is also observed how for even higher scan speed other defects occur, regardless of balling.

It is also observed how the thickness of the layer also affects balling with a positive correlation (the thicker the layer, the more difficult is to correctly melt all the powder, thus the easier the formation of balling is).

The combination of these three parameters is the volumetric energy density:

$$\varepsilon = \frac{P}{vhd}$$

- P = laser power;
- v = scan speed;
- h = scan line spacing;
- d = layer thickness;
- $\varepsilon$  = volumetric energy density (VED).

Low VED results in low working temperature in the lower part of the layer, generating a steep temperature gradient in the vertical direction. This causes splashes of the melt pool and therefore the second kind of balling, requiring an appropriate layer thickness and a VED regulated accordingly.

Finally, the powder composition is considered: composition with higher tendency to oxidation results in a lower wettability, leading to balling of the first kind. Oxidation is lower in powder with bigger grains, but the addition of traces of deoxidant also is proven to be effective.

*J. Bi - Effect of process parameters on formability and surface quality of selective laser melted Al-Zn-Sc-Zr alloy from single track to block specimen [5]*

A particular alloy (Al-Zn-Sc-Zr) is used to demonstrate the effect of process parameters in Selective Laser Melting, pointing out how pieces manufactured with SLM have great mechanical properties due to the high thermal jump to which they are subject.

The experiment was conducted with an Al-Zn-Sc-Zr alloy in a EOS M290 SLM machine, subjecting the powder to a combination of laser power and scan speed, while layer thickness is kept constant at 0.03mm and so is the hatching space at 0.08mm. The artifacts created are blocks of dimension 7x7x12 mm, while the characterization is performed by removing the artifacts from the base, chemically treated and the surface roughness measure by a scanning confocal microscope. Results are the average of three measurements.

Final results show how laser power and scan speed can be related by the expression of energy density  $w = P/v$ , and that the characteristics of the track are influenced by this. In particular for low laser power and high speed a decrease in the width of the track is observed, as well as in the depth of the track. This behavior is smoothed when both parameters are non-ideal (power changes with fixed high scan speed doesn't achieve as much change as power changes with fixed low scan speed does, and vice versa).

The effect on the morphology, observed with the microscope, is also positively correlated with high energy density, in terms of spreading of the melt pool and higher density. It is also observed, however, how excessive laser power lead to a disproportionate volume expansion, thus generating cracks with the subsequent contraction. Considering the scan speed, low speed is observed to allow a correct formation of the track, while an increase can lead to balling phenomena at the edge of the track, due to the increase in the solidification rate and the impossibility for the liquid to correctly spread.

Surface quality observations related high scan speed with defect formation such as balling phenomena and cracks. The roughness of the surface decreases with the decrease of the scan speed, due to the fact that for low speed the liquid metal has more time to spread evenly. At low speed, finally, the air porosity has more time to escape the melted pool, with a beneficial effect in term of density.

It is concluded that optimal parameters are those that ensure that high energy density is applied to the track, although a limit is applied to laser power to avoid cracks. Singularly, laser power and scan speed have an higher effect on the track dimensions than the energy density altogether, showing how different combination of the two can have a diverse effect: high laser power allows greater penetration in the powder bed, while the scan speed is greatly responsible for the width of the track, since it gives more time to the liquid to spread. A high energy density allows also a greater overlapping with the layer underneath and on the side. The comparison between results for different combination of parameters allows to find the optimal ones, with a process characterized with high repeatability for other material.

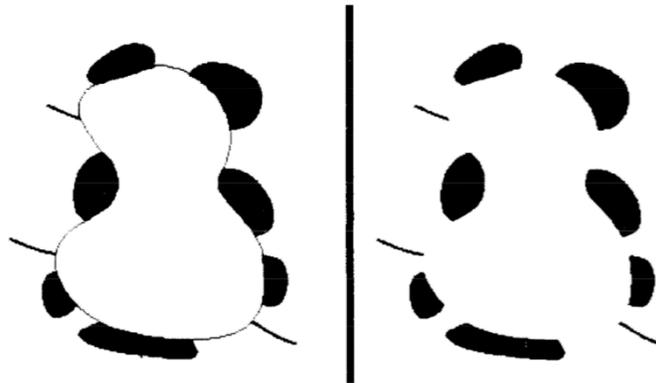
## Image analysis:

*M. Kass – Snakes: active contour models [6]*

A method for active contouring is described, by using a so-called snake, an energy-minimizing spline influenced by external constraints and image forces, proportionate through user-tunable parameters, with the goal to provide a method for active contouring that is not limited to one solution but can be customized by the user.

An energy functional is presented as the sum of three components: internal energy ( $E_{int}$ ), image energy ( $E_{im}$ ), constraints energy ( $E_{const}$ ). These three components are then singularly explained.

The internal energy regulates the behavior of the snake, with a first order component causing the snake to act like a membrane, and a second order component causing the snake to act as a thin plate, both multiplied by tunable parameters. In this context a GUI to test the tuning is provided (“snake pit”). The image forces cause the snake to be attracted by image features such as lines, edges and corners. The attraction to lines, for example, works by using as energy ( $E_{lines}$ ) the intensity ( $I_{(x,y)}$ ) of the image; the tuning of the correspondent parameter can generate repulsion or attraction depending on the sign. The energy component of the edges ( $E_{edge}$ ) is represented by a gradient of intensity multiplied by a Gaussian, thus attracting the snake to zero crossing edges. Applying again a Gaussian to smooth the image, the energy component for terminations ( $E_{term}$ ) is computed as a partial derivative of the gradient angle in the point with respect to the perpendicular direction. The application of both the edges and terminal energy causes the spline to fairly predict edges (fig. 2.6)



(fig. 2.6) the smoothness of the curve depends on the internal energy. Kass et al. [6]

The constraints energy is not explained in depth, due to its largely varying nature, but an example is given in presenting the interactive model in the form of elastic constraints (as springs between snakes) and a repulsion force originated in a specific point in the plane. An energy component to account for stereo effect is also applied, and finally an explanation on how the snake is able to follow edges in motion is given, with a frame-to-frame example.

$$E_{snake} = E_{im} + E_{const} + E_{int};$$

$$E_{int} = \frac{\alpha|v_s|^2 + \beta|v_{ss}|^2}{2};$$

$$E_{im} = w_{edge}E_{edge} + w_{line}E_{line} + w_{term}E_{term} = w_{edge}(-(G_\sigma \nabla^2 I)^2) + w_{line}I_{(x,y)} + w_{term}\frac{\partial \theta}{\partial n_\perp};$$

*C. Galambos – Progressive probabilistic Hough transform for line detection [7]*

An algorithm is presented to perform a Hough Transform improving the already existing probabilistic approach. At first the previous algorithm is described: the probabilistic Hough Transform is a randomization of the standard Hough Transform, performed by using the HT to select a number of inputs points and use the set for the voting procedure instead of the whole pool of points available. The limitation of this approach is the necessity of some a-priori knowledge to determine which points belong for sure to the line, thus making the approach not very feasible in countless real applications. The presented PPHT works by repeatedly select points for voting, and after a certain amount of points have voted, sample the resulting poll many times. The samples are then analyzed to check if a certain threshold of votes is exceeded, a threshold associated with random noise. If a sample has a number of votes above the threshold, it is considered an identified line. The liability of this process is the false positive, since all the lines detectable through HT are also detectable through PPHT, in principle. A precise outline of the algorithm is presented, point by point, then a comparison is performed on different images. Accuracy's results are evaluated subjectively, while computational improvements are evaluated by number of vote casted, where the logic is similar, and CPU time where there isn't a common parameter that can guarantee an accurate comparison. The PPHT is comparable to the HT in accuracy, with different behavior that are difficult to categorize as better or worse (for instance HT tends to detect long lines whereas PPHT tends to translate the info in multiple shorter lines). The main difference between the proposed approach and the existing ones is the computational effort: for simpler images PPHT casts almost a fifth of HT's votes, and takes half the time as other approaches that are compared in CPU time. For more complex images the votes' ratio is similar, while the CPU time is a sixth as for other algorithms. It is highlighted how this greater efficiency with similar result is promising for applications where a quick response is needed.

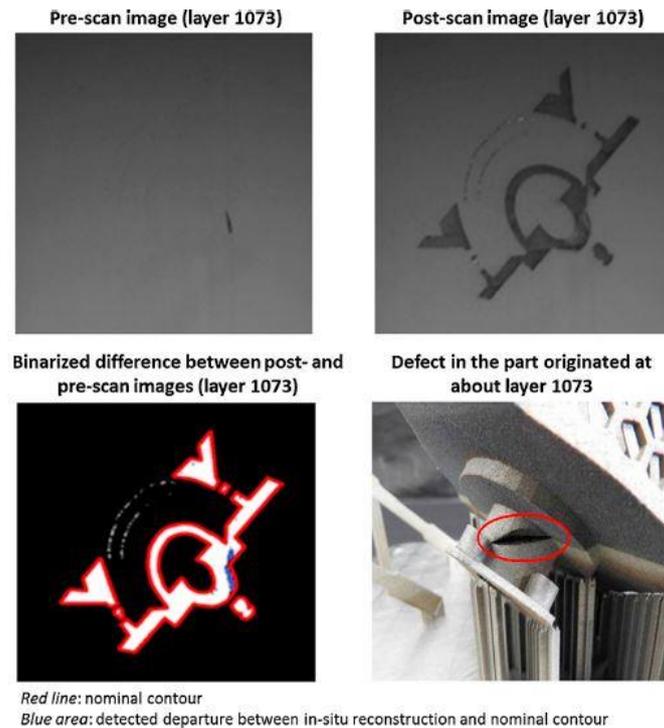
*F. Caltanissetta – characterization of in-situ measurements based on layer wise imaging in laser powder bed fusion [8]*

A method to detect deviations from the correct manufacturing process is proposed, exploiting an off-axis camera to acquire images of the artifact before and after each layer is posed, thus recognizing associating visual features to defect formation to allow the recognition of said defect before the occurrence.

A particular attention was reserved to the acquisition conditions, as the availability of an experimental apparatus allowed to precisely control each variable, from the building parameters to the environmental conditions, such as inert atmosphere, lighting conditions and design of production. The control of lighting conditions, in particular, allows to characterize their influence on image analysis.

The approach to the analysis follows the use of active contours for segmentation, analyzing its advantages and drawbacks as well as those of edge detection. In this experiment a-priori knowledge of the model is used to initialize the contour, then a preprocess is developed to enhance the foreground from the background to contain the weakness of active contour against inhomogeneity in the brightness.

## Image analytics for in-situ defect detection in Additive Manufacturing



(fig. 2.7) propagation of defects can be detected with layer by layer monitoring. Caltanissetta et al. [8]

The actual acquisition is performed once the manufactured layer is cooled down, to account for shrinkage deformation. After the in-situ acquisition the artifact is removed and an ex-situ acquisition using an optical microscope is performed, to allow comparison and thus validation of the in-situ measurement as identification method of visual features. The results of the comparison validate the robustness of active contouring in laser powder bed fusion acquisitions expressing the geometrical error in around  $100\mu\text{m}$ . The effect of brightness is measured computing the Dice's index for segmentation techniques on both the original image and the transformed image. The results show how except for one case with low illumination in the front (resulting in a darker but more even pixel intensity) that showed from the start a high correspondence, the transformation improve significantly the Dice's coefficient. Incidentally the dark field case is the one with higher accuracy, although the implementation of the transformation produces results comparable, with a lower accuracy in bright field cases. The experiment is carried out comparing the usage of active contouring with a methodology from the literature, concluding how the produced measurements are similar to each other, while computationally the active contour model can produce an accurate result in few iteration, as for the compared method, but executing each iteration much faster. It is pointed out how this is appealing for near real-time application.

## Machine Learning for Additive Manufacturing:

*J. Alldredge - In-situ monitoring and modeling of metal AM powder bed fusion [9]*

The paper targets the issue of having highly costly hardware and heavy computational demands for sensing and feedback control in additive manufacturing. The proposed solution is a low-cost thermal camera combined with a faster-than-real-time qualitative model to detect anomalies enabling corrective actions or the complete stop of the process. The exploited camera allows a relative temperature measurement, allegedly sufficient to identify the relevant features. A distinction is made between physic-based models and in-situ sensing, with the latter being better fitted for real-time application due to lower needs for computational resources. The conclusion of the paper is the validation of a low-cost solution to detect physical variation, through a comparison with previous experiments done with high resolution thermal camera.

*M. Grasso - A statistical learning method for image-based monitoring of the plume signature in laser powder bed fusion [10]*

The paper proposes a statistical learning method to correctly separate cluster of features and apply an early diagnosis in laser melting manufacturing. The laser heating zone and the plume resulting from the partial material vaporization are regarded as signature of the quality of the process, referring to previous literature that perform such an association.

The recording device is an infrared camera, hence the impossibility to distinguish the heat trace of the laser heating zone (LHZ) and the plume. Both are then considered as region of interest (ROI).

The paper proposes to identify as a characteristic of defect presence, the out of range ROI.

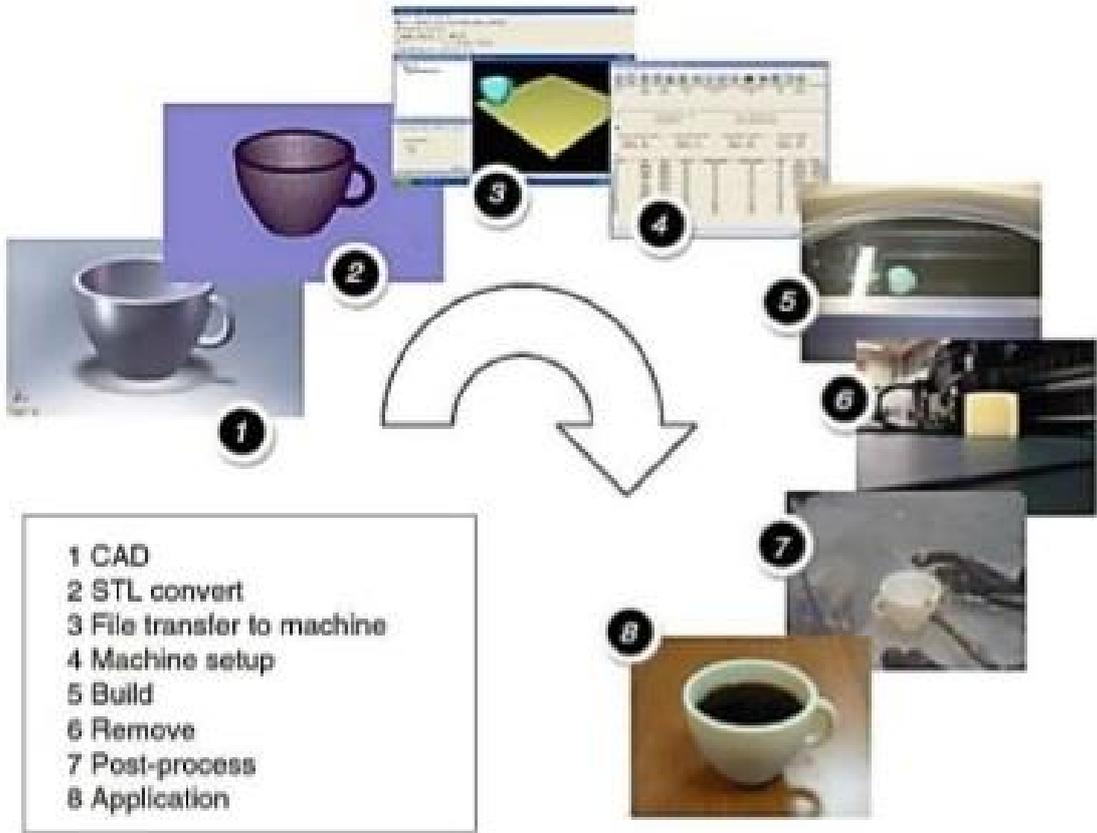
The challenge presented is the contemporary presence of the LHZ and the plume, thus not allowing a simple recognition (the dimension and shape of one of the two could be mistaken by the algorithm as a defect symptom). This is called “multimode” process, in which the “normal condition” can be one of multiple state. To face this challenge a nonparametric method is proposed, K-chart, based on one-class classification SVM called SVDD (support vector data description).

Due to the fact that input data can be represented by two feature (area and distance from the origin) and that only a combination of the two represent spattered material (considered a sort of noise), K-chart methodology is also used for preprocessing, in order to correctly classify the data. Subsequently the ROI is extracted through image analysis and a statistical description of it allows another application of K-chart to define if the process is faulty or not.

The conclusions show how the application of nonparametric statistical method allows to better represent the process real conditions in a multimode process, and how the plume can be used to perform early fault diagnosis.

### 3. Process description

The term “Additive Manufacturing” (AM) indicates the manufacturing process based on the increase of the build volume instead of material subtraction as in traditional means of production. The selected material is added layer by layer in a different fashion depending on the technology and the material, and different constraints are applied to both the environmental conditions and the building strategies.



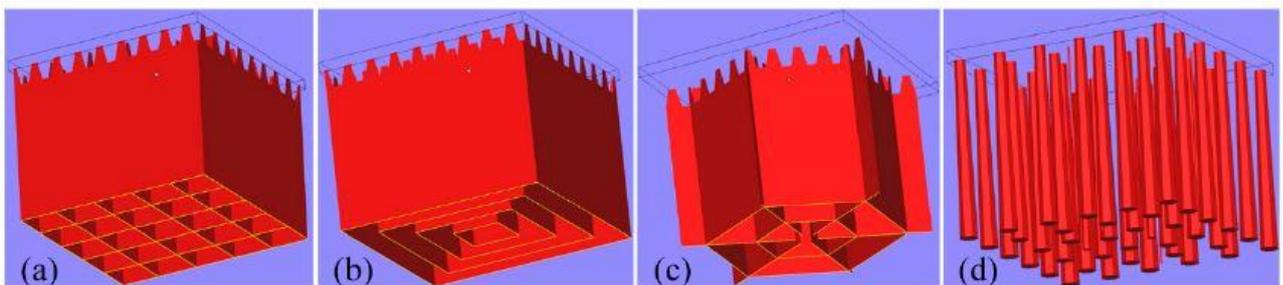
(fig. 3.1) traditional steps for AM production. Gibson et al. [12]

## Techniques differentiation:

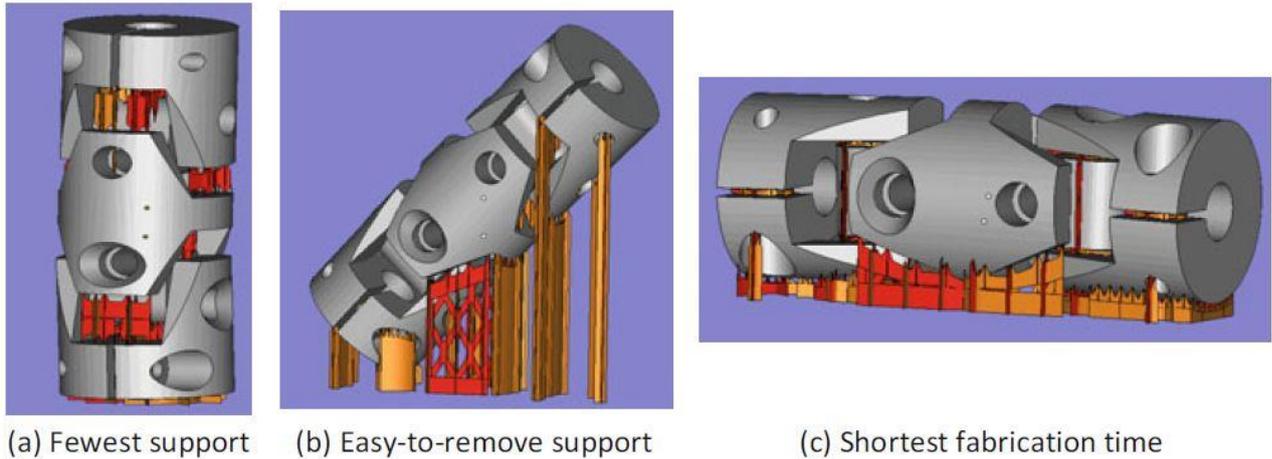
Born around the middle of the 20<sup>th</sup> century, the additive manufacturing began to be used for rapid prototyping in 1987 with the entering in the market of the Stereolithography Apparatus (SLA-1) [11], and thrived in the last two decades with the commercialization of different technologies from various company.

The very basic principles of AM allow improvements with respect to the traditional manufacturing industry. The most obvious difference is the saving of material, since no longer the process implies the subtraction of material from a greater volume to reach the desired shape. At the same time, no specific set of tools is needed to create the object, with the exception of the ones involved in the normal functioning of the machine. This means that contrary to processes like turning, where different tools are needed for different parts or different objects, in AM no customization is needed. Another key improvement is the time saving character of this technique, not because of the build time in itself, as much as because of the reduction in the number of operations needed.

Traditional manufacturing often requires different steps and different techniques to produce a finite piece, whereas AM always relies on the same chain, highlighted in (fig. 3.1) [12]. The first step of this chain is the CAD of the piece, a phase also affected by the means of production as the pieces are subject to different constraints with respect to traditional manufacturing. For instance, more articulate shapes are possible, since there's no tool's volume to consider, but since the construction happens one layer at a time is also necessary that the layer in construction can rely on a foundation. Therefore, supports are needed under overhanging areas and unsupported areas, in a configuration that optimize easy removal in post processing, fastest building time, and a minimization of the supports themselves (fig. 3.2, fig.3.3), also depending on the technology. In principle, the design should also take into account the so-called "staircase effect", the thickness of the layers is usually lower than 50 $\mu$ m, nonetheless a certain limit on the border distance must be respected in the design phase. [11]. Once the CAD is completed, another file is created; this operation of slicing translates the original CAD in a file that distinguish different layers, and allows the machine to execute scan strategies, so the next step is the upload of the file to the machine and the manufacturing of the process. The scan strategies should also account the thermal gradient in the piece, since the nature of AM implies that the material is subject to sudden changes in temperature: it's important that the material doesn't cool down too much during the process or the next layer will cause a large temperature gradient. The path of the heat source (being the material itself or a laser applied to the material) must be decided accordingly. Finally, postprocessing operations are performed to remove the supports and reach a satisfactory quality (by exploiting mechanical or chemical properties of the material).



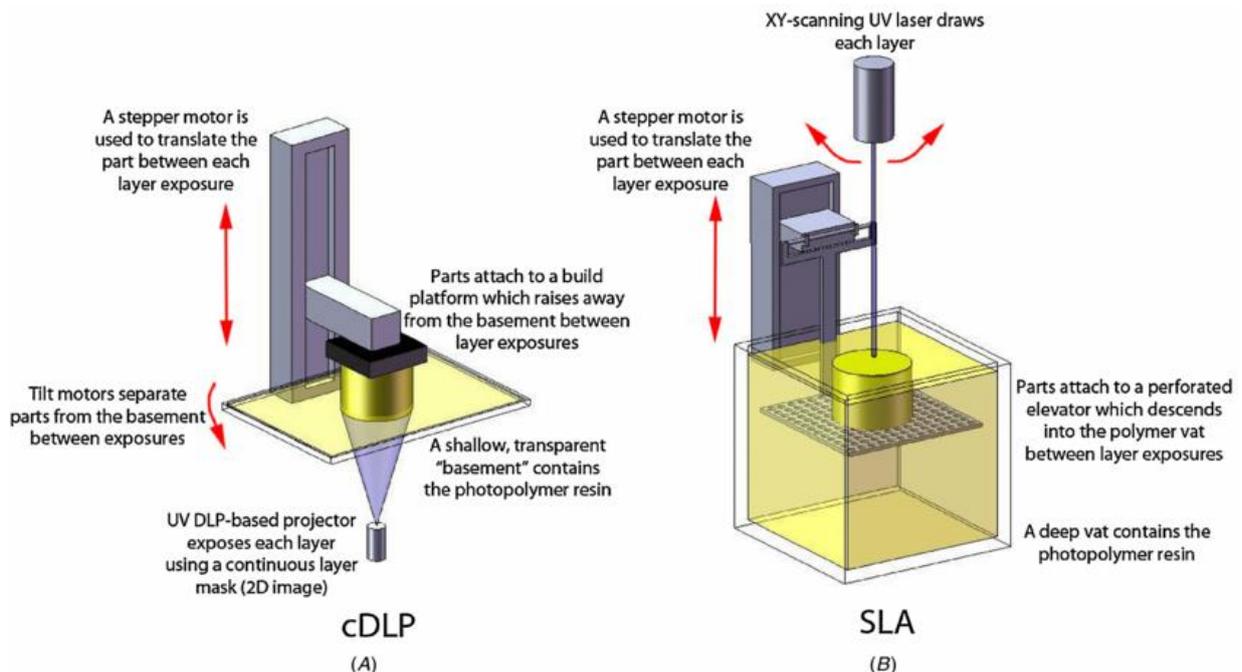
(fig. 3.2) designs for supports: (a) block; (b) contour; (c) web; (d) cone. Zhang et al. [22]



(fig 3.3) tradeoff between different configuration of supports. Yang et al. [11]

Different materials are already implemented, such as metallic powder, polymers, ceramic, and biological material. While the same principle of adding material one layer at a time is applied, different techniques are required accordingly to the material used.

One such example is the aforementioned stereolithography, example of building process that start from a liquid component to generate the 3D object. Materials used in this application are photopolymer resins, namely polymer that starts in a liquid form and can be solidified by applying UV light, to which the material is sensitive. The principle for this process is the polymerization, a chain reaction that turns a liquid mixture of monomers into a defined polymeric string. In stereolithography the polymerization is perform accordingly to the desired shape of the layer, that defines the scan strategy. Two possible options are available to switch to the following layer, the continuous digital light processing (cDLP) and the SLA (fig. 3.4).



(fig 3.4) difference between continuous Digital Light Processing and Stereolithography Apparatus. Wallace et al. [20]

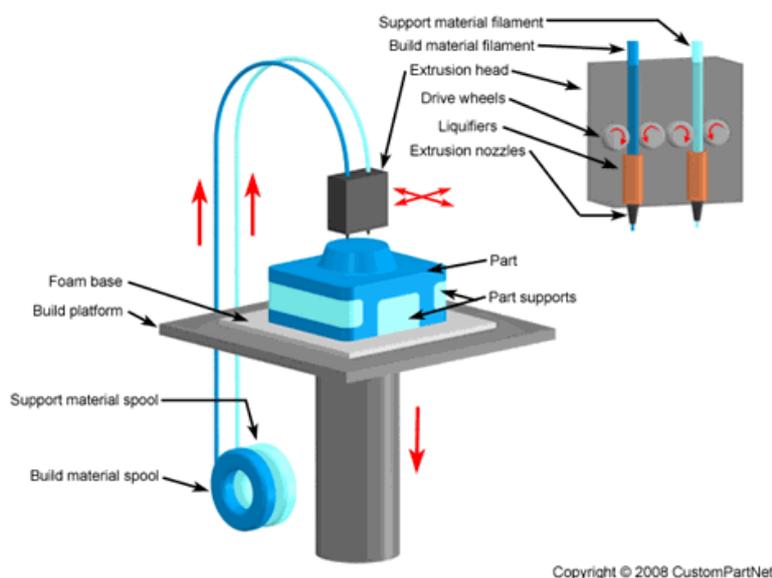
## Image analytics for in-situ defect detection in Additive Manufacturing

In the first the resin is contained in a thin, transparent plate, that is refilled constantly, while the UV projector creates the layer from below the plate. The object is attached by the first layer to a platform that rise during the process, “extracting” the piece while it’s been manufactured. In the second configuration a platform is posed on the surface of a tank filled with resin, and as the layer are manufactured through a UV laser, the platform dives deeper into the tank, leaving only the last layer near to the surface. The final result will be the complete piece inside the tank full of resin.

Thanks to the fact that stereolithography was the first technology commercialized, extensive literature exists that analyze the physic around the polymerization process and the energy distribution in the process [13].

The second kind of application covered is the extrusion methodology, consisting of preparing the material before the deposition and then depose the material on a section to create the 2D layer, the section being the support plate or the previous layer. A technology that exploit this application is Fused Deposition Model (FDM).

In FDM a thermoplastic polymer is stored as filament and connected to the machine through a hose with a heated end (fig. 3.5), so that the polymer in exit melt and can be applied to the layer as a melted 1D track. The final product of FDM is a polymeric object that has, due to the filament shape in which is provided, an anisotropic behavior in terms of mechanical properties. In particular, along the filament length the strength is higher than in the other directions. As for stereolithography, extensive tests have been performed on this technology: the anisotropic behavior just mentioned was characterized in [13] and the ratio between tensile strength along the filament deposition direction and both the perpendicular on the same 2D plane and the vertical direction is estimated to be in the range of 10% to 65% (express by the so-called “strength isotropy factor” in [11] in the range 0.1 - 0.65), depending on the process parameters and, more importantly on this account, the environmental conditions, such as chamber temperature and the hose end temperature, that heavily influence the merging between the cooled pre-existing track and the new layer. This means that environmental condition plays an important role in the formation of an optimal product, thus posing technological requirements to the machines.



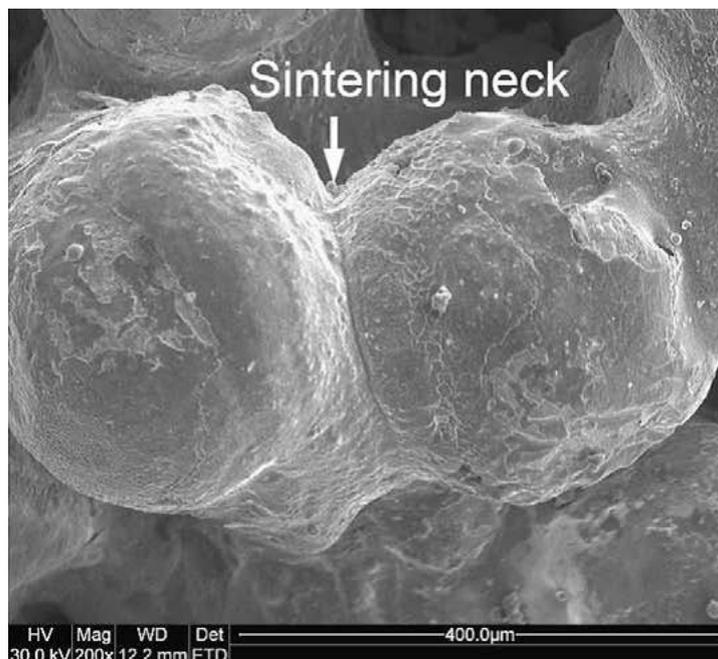
(fig. 3.5) FDM schematic highlights the usage of different material for the supports. CustomPartNet [23]

Machine based on material's extrusion present a very promising solution for simple manufacturing, that in recent years has seen a widespread commercialization not only for bigger industrial solutions, but also for home kits, opening possibilities for innovative logistic of production and object transferring.

Another widespread AM technology is metal additive manufacturing. As the name suggests, the material used in manufacturing is a metallic powder, lay down on a building platform and subject to an aggregating process to create the solid piece (fig. 3.6). This process is mainly the action of a heat source like a laser or an electron beam, applied to each layer; in addition, the power needs to be enough to guarantee the adhesion between subsequent layers, and, at the beginning, between the platform and the first layer. Contrary to the solutions seen before, metal additive manufacturing is a branch of AM that intrinsically encounter greater difficulties in being commercialized for the end-user, mainly because of safety concerns, since the heat generation (being laser or electron beam) needs to work on metal material that requires higher powers, thus requiring much more careful handling. Other risk factors are related to the usage of metal powder (that requires protective equipment) ranging from irritation to explosion risk, exacerbated by the environment requirement for production. [16, 19].

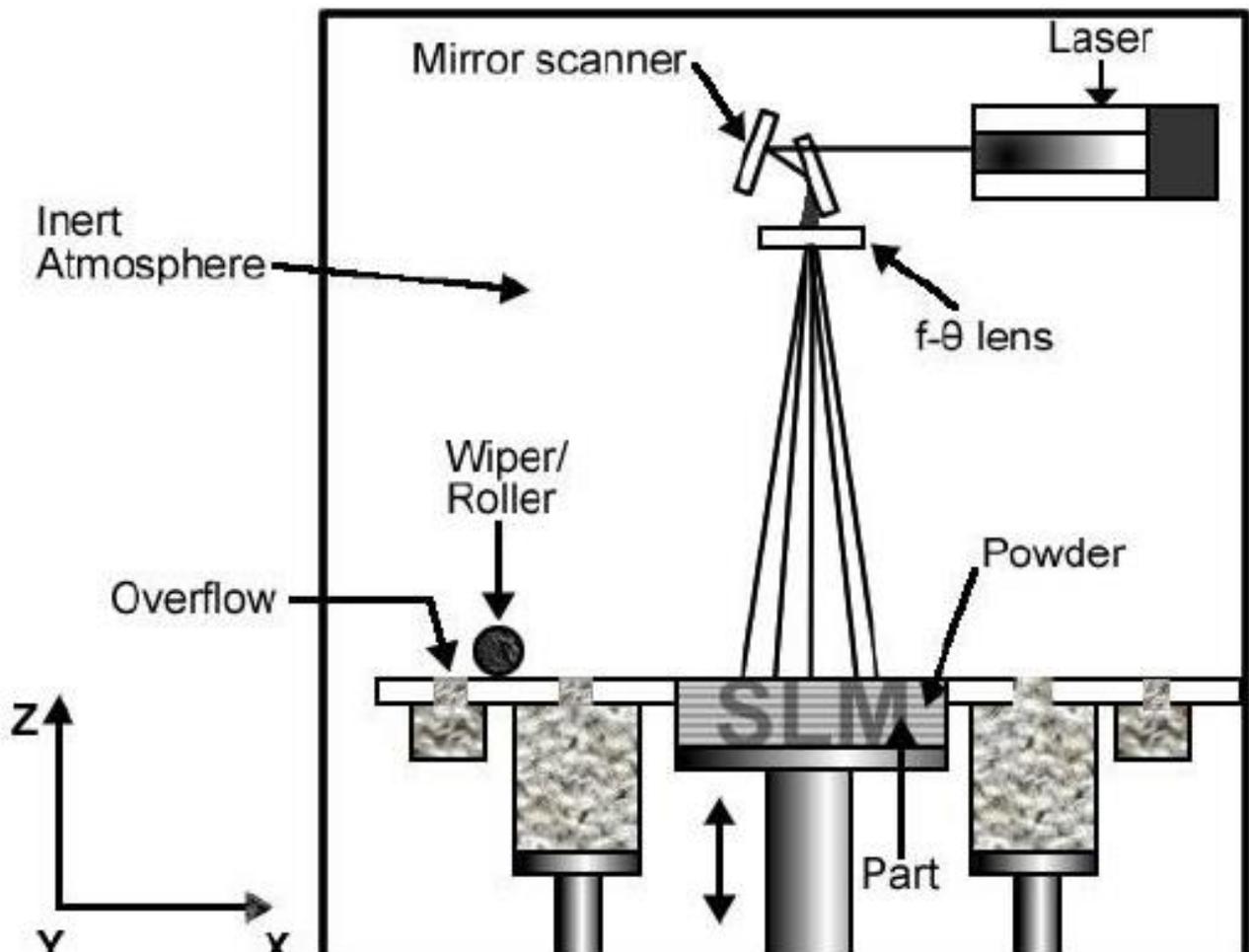
Three main applications have been developed, Selective Laser Sintering (SLS), Electron Beam Melting (EBM) and Selective Laser Melting (SLM). A brief description of these will follow.

SLS produce pieces with high yield strength and is commonly used for rapid prototyping and production of specialized part. In the building process, the powder is spread on a platform one layer on top of the other, and a high-power laser scan the desired section to liquefy the surface of the powder and allow the cohesion of close grains without reaching complete melting (sintering is by definition "the welding together of small particles of metal by applying heat [or pressure] below the melting point" [15]). This adhesion is characterized by lower density than complete melting due to the partial retainment of the form by the grains, thus binders with low melting point are often used. These, however, usually have different properties than those required, so in post process the binder might need to be removed.



(fig. 3.6) the cohesion between grains in SLS is called necking. Gu at al. [4]

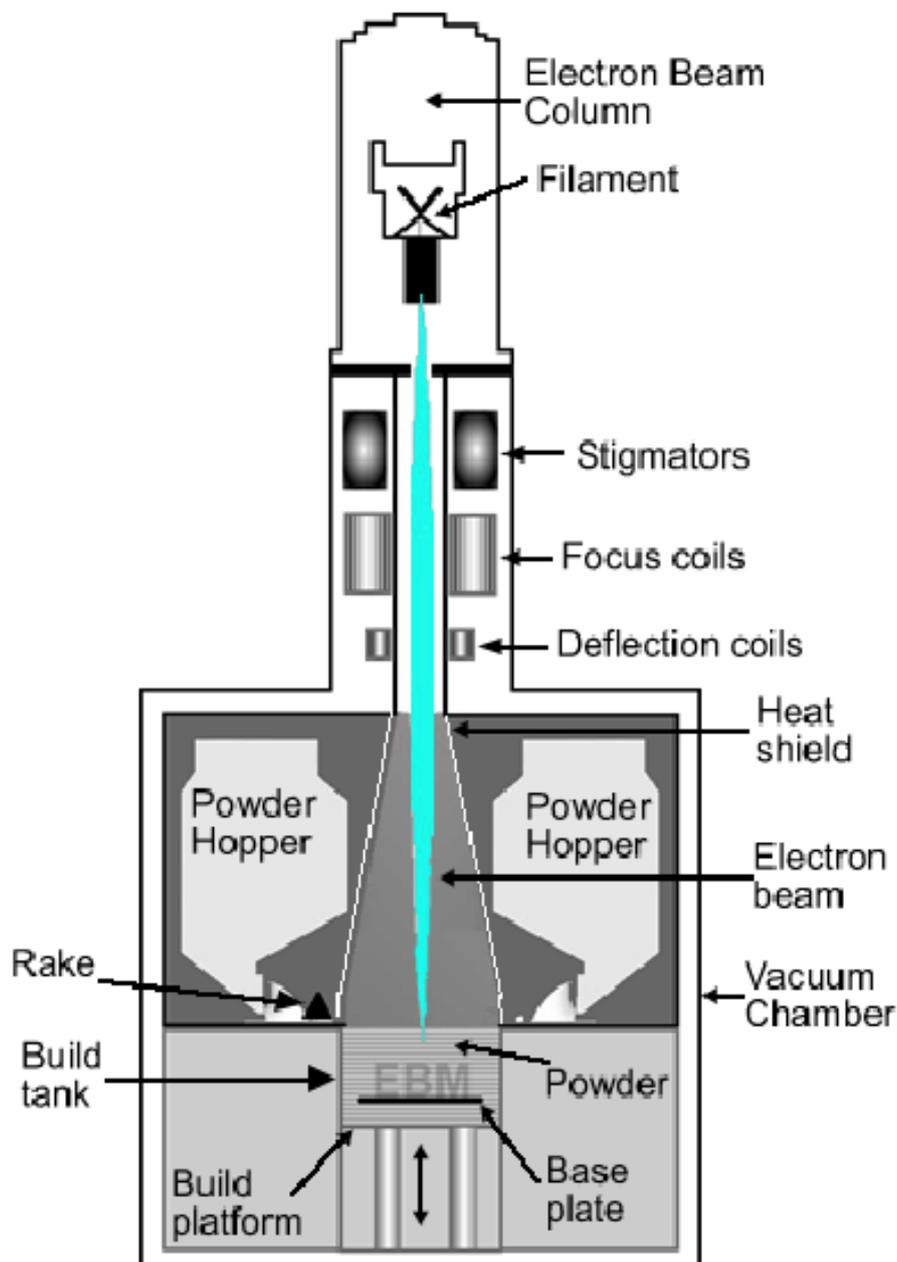
SLM (fig. 3.7) is also based on laser for heat generation, but contrary to SLS, complete melting is achieved, allowing greater density without postprocessing except for superficial precision and removal from the base plate. Density of the piece is guaranteed by optimal parameters such as hatch distance, laser power and speed, but also aforementioned environmental conditions. Building environment, in fact, is subject to control to different parameters, such as temperature, oxygen concentration and inert gas presence. The control of the air composition is performed to ensure both production quality (due to the high reactivity of the oxygen), and safety, since the reactivity of the compounds usually present in the air might cause violent reactions. From the perspective of manufacturing quality, chamber temperature isn't less important, as it plays a key role in ensuring that the thermal gradient to which the powder is subject is maintained as low as possible, alongside a wise choice of scan strategies. The simplicity of building, the chance to achieve more complex geometry and to save time in tool application are the reasons for which SLM, although in itself time consuming due to the need to create thin layers for quality, is used for production of prototypes and small series parts, but also in sector in which high mechanical properties are required for customized parts (such examples are the usage of SLM for medical prosthesis, dentistry, aerospace applications).



(fig. 3.7) SLM schematic highlighting the mirror apparatus necessary for the manufacturing. Sidambe A. [24]

EBM relies on a process very similar to that of SLS/SLM, except the heat to achieve melting is generated by a concentrate electron beam (fig. 3.8). The key difference in the processes is that the higher atomic mass of the electron with respect to the photon allows greater fusion power to be transmitted, allowing for different mechanical characteristic to be attained, as reviewed in [17]. As for SLM and SLS, environmental conditions are a key factor to block interaction between the beam and the atmosphere, thus high vacuum is achieved (greater than  $10^{-3}$  Torr), although in the process described by *Murr at al.*, helium is injected to enhance heat conduction, increasing the pressure to  $10^{-2}$  Torr.

Compared to the early technologies like stereolithography, metal additive manufacturing is fairly new, fueling researches on parameter optimization, material characterization, applications, and control strategies.



(fig. 3.8) EBM schematic. It's evident the different machinery's volume with respect to SLM. Sidambe A. [24]

## Defects analysis in Selective Laser Melting:

Defects in SLM occur mainly because of non-ideal process parameters or raw material quality issue. The latter issue regards the density of the powder before the process. Namely, a low-density powder can cause the gas in between to remain trapped in the artifact during the formation of the melting pool, since the extreme cooling rate doesn't allow the bubbles to escape the melted pool. This **porosity** results in lower fatigue resistance, even if it has a smaller impact compared with other defect influence (porosity leads to spherical defect, that are proven to be less influential on the fatigue resistance under a certain concentration), but at high enough concentration can lead to a decrease in tensile properties [18]. Porosity can also be the result of too high energy density applied in the process, that result in a vaporization of some components of the alloy, generating the gas bubble trapped in the melted pool.

Non-ideal process parameters also generate other defects, related to **incomplete fusion holes** (also called LOF, Lack of Fusion). This category is caused by an energy density too low, which results in some area where the fusion is incomplete, thus leaving powder in some area of the artifact or poor bonding between material portions. LOF defects are more likely to presents themselves in the border regions of laser tracks or between layers (in case the laser power isn't enough to correctly melt the lower strata. They result in a worsening of tensile properties and have a great effect on fatigue resistance of the piece.

The working conditions are, themselves, a possible cause for defect formation: high temperature differences and the cyclic dilatations and contractions, for instance, may induce the material to exceed its elastic deformation limit, thus generating high internal tensions, especially in alloys with high thermal expansion coefficient and low thermal conductivity. This can result in **cracks** and compromise the fatigue duration of the piece.

Finally, is worth mentioning a phenomenon called **balling**, caused by an incorrect selection of scan speed and laser power [4]. Balling is an issue in which the melted pool isn't completely smooth, preventing the powder from forming the so called "sintering neck", a connection between conglomerates of materials. The final result is a series of spherical imperfection instead of a continuous and smooth track. This issue can be caused by two conditions, namely a laser power too low in an ideal scan speed condition (resulting in imperfection approximately the size of the laser beam caused by limited liquid formation) or a scan speed too high with a sufficient laser power (resulting in a high number of small-sized balls induced by melt splashes).

Defects can also occur due to process errors, mostly during the powder deposition. These will be described in depth during the description of the machine working principle.

To avoid oxidation, the atmosphere of the chamber must be controlled, either with vacuum or inert gas: the machine used in this thesis project uses Argon to fill the chamber. Moreover, a big part of the literature on AM is related to finding the optimal machine parameters, combining the right laser power and scan speed, as well as track definition (see "Effect of scan strategy" in [8]), with correct environment conditions.

The machine used in the thesis project, EOS MS290, use a patented method to prevent porosity, other than the tuning of the parameters: the employment of a gas flow ("air" would be incorrect, given the chamber controlled environment) to blow the smoke resulted from the laser action in a direction that guarantee it will not meet the laser, situation that could lead to occurrence of porosity [19].

## 4. Infrastructure description

The thesis project involved the use of a SLM machine in its normal working conditions, and a camera used to capture images of the working process. The camera was originally supplied with the machine, thus the necessary components to install the camera were already in place, such as the support bracket and the “window” in the upper part of the chamber to allow camera acquisition. The software version, however, did not include the camera control, that is the main reason for the development of a capture script to be run by an embedded board. A correct understanding of the process in its whole is deemed necessary to exploit the machine characteristics in order to acquire the images at the optimal moment. For this reason, the machine paragraph also contains a sketchy description of the building process common in selective laser melting.

### EOS M290 Laser Melting system for metallic powder:

The SLM machine used in this project, as aforementioned, is a Selective Laser Melting machine produced by EOS GmbH, world leader in laser powder bed fusion systems (fig. 4.1). Some technologies hereby described are currently patented by EOS, thus not applied to the generic selective laser melting technology. The material used for manufacturing is AlSi10Mg to produce pieces of different dimension.

EOS M 290	
Building volume	250 mm x 250 mm x 325 mm (9.85 x 9.85 x 12.8 in) (height includes build platform)
Laser type	Yb-fiber laser; 400 W
Precision optics	F-theta-lens; high-speed scanner
Scan speed	up to 7.0 m/s (23 ft./sec)
Focus diameter	100 µm (0.004 in)
Power supply	32 A
Power consumption	max. 8.5 kW / typical 3.2 kW
Nitrogen generator	integrated
Compressed air supply	7,000 hPa; 20 m³/h (102 psi; 706 ft³/h)
<b>Dimensions (B x D x H)</b>	
System	2,500 mm x 1,300 mm x 2,190 mm (98.4 x 51.2 x 86.2 in)
Recommended installation space	min. 4,800 mm x 3,600 mm x 2,900 mm (189 x 142 x 114 in)
Weight	approx. 1,250 kg (2,756 lb)
<b>Data preparation</b>	
Software	EOS RP Tools; EOSTATE; EOSPRINT; Materialise Magics RP with SG+ and further modules
CAD interface	STL. Optional: converter for all standard formats
Network	Ethernet

(fig. 4.1) M290 datasheet [25]

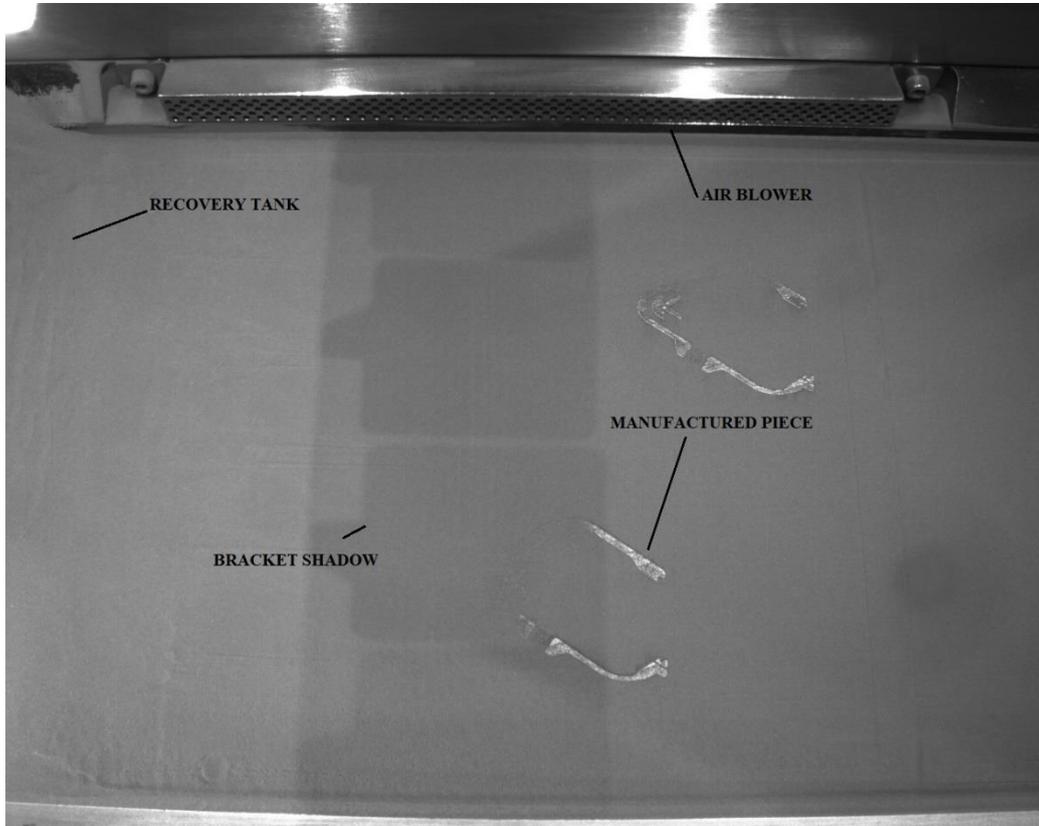
As described in the chapter “Process description”, selective laser melting requires specific environmental condition to achieve a satisfactory manufacturing and control the process safety liability. In particular the raw material belongs to the metal powder class B, subject to fire and explosion hazard, and health hazard due to contact or ingestion [19], thus requiring the lowering of the oxygen percentage to reduce the reactivity: this is achieved by removing the air in the chamber and injecting in its place an inert gas until the oxygen percentage is below a safety threshold. Moreover, to achieve the desired quality of the melt pool and therefore of the piece, the building site is preheated before the job.

The machine is composed, in a nutshell, of the following components:

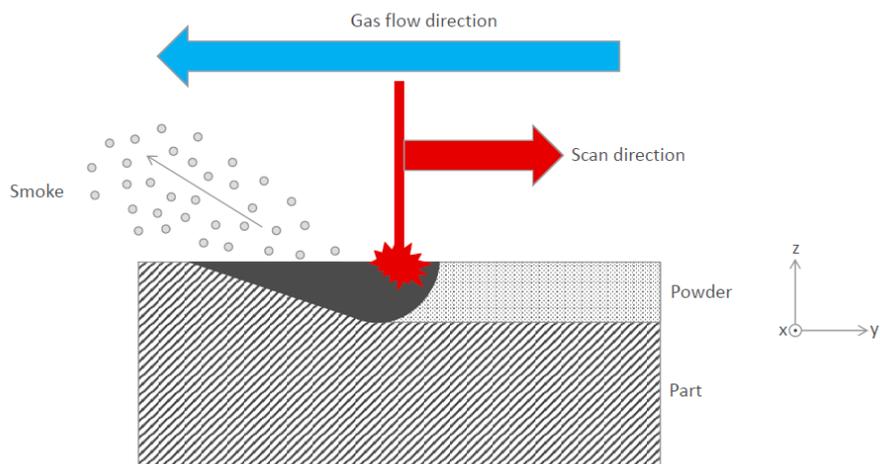
1. Chamber (fig. 4.2): a controlled environment in which the pieces are produced. It is composed of a building site, a mechanical recoater to move the powder, two tanks to store the powder, and an air blower studied to prevent balling effect due to the gas produced in the melting. It has an access in the front of the machine to perform manual operation (powder reloading, collecting of the product, maintenance), and two windows on the roof to allow image acquisition and to give the laser access to the chamber;
  - a. Recoater: a mechanical arm that moves left-right to distribute the powder from the powder tank to the building site. The quantity of the powder is regulated by the vertical movements of the powder storage tank;
  - b. Building site: a platform in which the powder is placed to be melted by the laser. Between the platform and the powder is fixed a metallic plate whose surface will be partially melted by the laser to fuse with the products. The platform descent vertically of a distance equal to the thickness of the layers in order to make room for the powder brought by the recoater;
  - c. Powder tank: a platform placed on the right with respect to the building site in which the metallic powder is stored before the start of the cycle. It can ascend vertically to let the mechanical recoater swipe the powder on the building site. The height which it moves, however, can be less or more than the layer’s thickness, in order to guarantee a better distribution of the powder when the operator decide that is necessary by providing more powder, or to save powder if an alert is raised that the powder might be insufficient. The percentage of powder with respect to the nominal quantity given when the movement is precisely one layer’s thickness is called dosing factor.
  - d. Recovery tank: a platform usually placed lower than the building site, on its left. It collects the extra powder after the recoating, that will be reused in another job.
  - e. Laser regulation bracket: this bracket is fixed to the chamber’s roof, and it is made of reflective metal, so the laser system can determine its own position by catching the reflection of the laser as a reference point;

## Image analytics for in-situ defect detection in Additive Manufacturing

- f. Gas blower: a patented system to prevent the interaction between the laser and the smoke generated by the heat, since this would result in a deviation of the laser and ultimately in a lowering of the quality of the track, if not defect formation (fig. 4.3);



(fig. 4.2) the frame will be later cropped to exclude the chamber infrastructure



(fig. 4.3) schematic of the gas flow implementation

2. Laser system: it works by producing a high intensity laser pointed horizontally on top of the machine, and then directing it through a mirrors system on the desired plane's coordinates of the layer. The mirrors movements are controlled to accurately match the laser speed set in the parameters;

3. Gas injector: it fills the chamber with an inert gas after the air is removed. The percentage of oxygen is constantly monitored, and the machine stops if the correct working conditions aren't met;
4. Heating system: it preheats the chamber to reach the working conditions in which the laser melting can successfully happen. Until the target temperature isn't reached, the machine remains in an idle status;
5. Illumination: a series of white light are installed on the chamber's roof, illuminating the working plane. The illumination condition will be of pivotal importance for image analysis, as the management of areas with different intensity will be required;
6. Electrical panel: where all the drivers and electrical connection are done. It will be key to give the embedded board signal of the different working points, as through schematic is possible to identify the drivers corresponding to components whose activation separate the working states of the machine;
7. External components: these external accessories such as aspiration hoses, are needed to empty the chamber of all the remaining powder and load the new material, or to store the inert gas to feed the machine.

The machine works as follow:

- Preparation: the powder tank is lowered to the lowest and filled with new powder from the external tank, the plate is fixed to the building site, the central piston is moved until the plate's surface reaches the zero z-coordinate, the recoater is moved to the extreme left, and finally the chamber is locked. The machine then prepares the environment by removing oxygen and injecting inert gas (Argon), and by preheating the plate until the target conditions ( $T=165^{\circ}\text{C}$ , concentration of  $\text{O}_2=0.10\%$ ) are reached, then it waits for the user to start the job;
- Layer deposition: to pose a layer the recoater moves from left to right past the powder tank, then the tank itself lift of the height correspondent to the amount of powder of a layer times the dosing factor, and the recoater moves back from right to left posing the layer on the plate. In the process the recovery tank is kept low to allow the powder to fill it;
- Laser action: once the powder is in place, laser is activated to melt it accordingly to the slicing model;
- Postprocess: at the end of the process the pieces are let cooled down until the heat hazard is manageable with personal protection devices, and the gas concentration is reduced below risk threshold.

If preparation and postprocess are performed once at the beginning and at the end of the process, layer deposition and laser action are repeated layer by layer until the piece is completed according to the model.

During the working process, the laser may need to perform a calibration. To do so the regulation bracket is exploited. This bracket also projects a shadow on the working plane as a secondary effect, possibly affecting the image analysis (fig. 4.2).

## IDS IU-1540SE-M-GL camera:

The camera exploited for image acquisition is an IDS IU-1540SE-M-GL (fig. 4.4) [26]:

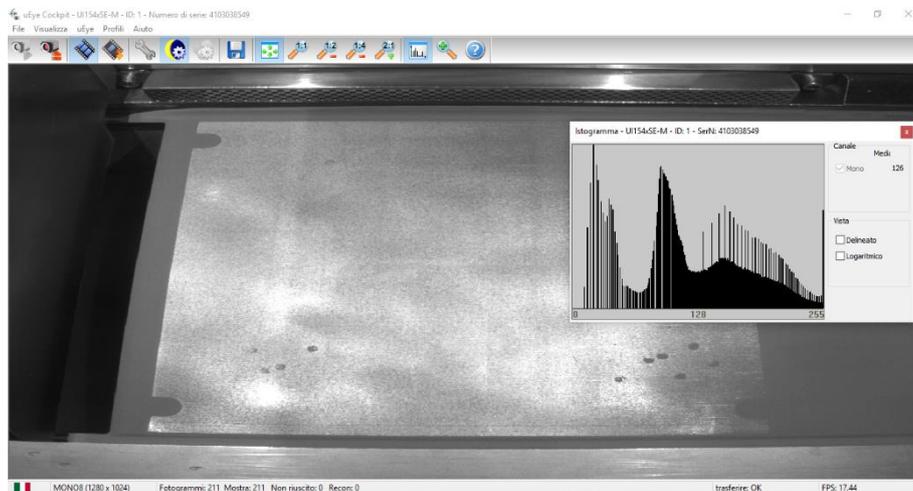
Sensor type	CMOS Mono
Shutter	Rolling shutter
Sensor characteristic	Linear
Readout mode	Progressive scan
Pixel Class	SXGA
Resolution	1.31 Mpix
Resolution (h x v)	1280 x 1024 Pixel
Aspect ratio	5:4
ADC	10 bit
Color depth (camera)	8 bit
Optical sensor class	1/2"
Optical Size	6.656 mm x 5.325 mm
Optical sensor diagonal	8.52 mm (1/1.88")
Pixel size	5.2 µm
Manufacturer	ON Semiconductor
Sensor Model	MT9M001STM
Gain (master/RGB)	13x/-
AOI horizontal	increased frame rate
AOI vertical	increased frame rate
AOI image width / step width	32 / 4
AOI image height / step width	4 / 2
AOI position grid (horizontal/vertical)	4 / 2
Binning horizontal	-
Binning vertical	-
Binning method	-
Binning factor	-
Subsampling horizontal	increased frame rate
Subsampling vertical	increased frame rate
Subsampling method	Color
Subsampling factor	2, 4, 8



(fig. 4.4) on the left, the camera datasheet, on the right the camera appearance [26]

The camera is fixed on the preexisting bracket on top of a circular window over the process chamber, and it is supplied with a USB cable in the electrical panel. This will be used to control the camera from a pc. It is possible to trigger the camera from an external input, in particular, python API are released from the producer, and a library is present: pyuEye. This allows to create a python script to perform acquisitions.

A software is also released, although not for embedded board. This will be used to perform a live monitoring of the camera and thus adjust the inclination of it in order to center the image (fig. 4.5). Since the acquisitions will be later cropped, what is needed in this stage is taking the whole building site inside the frame, adjust the focus of the camera, and then keep the camera fixed. This way the crop that will be performed during the analysis stage won't need dynamic parameters.



(fig. 4.5) uEye cockpit, the interface that allows the monitoring of the process chamber.

## 5. Acquisition script

As aforementioned, some python API are available to communicate with the camera, thus a python script is developed to handle the image acquisition. The main idea is to create an algorithm that activate the camera to capture images of the building site, layer after layer, exploiting a hardware infrastructure that will be developed in parallel to the software.

Although the hardware setup will be covered in a different chapter, the various steps will be referred to as to provide a context to the code evolution.

For prototyping, a Raspberry Pi 3B+ is used, chose mainly for versatility. Once the prototyping is over, however, a more robust and precise setup is installed, using a pc and an Arduino UNO for I/O management.

### Raspberry Pi Version:

The first operation performed is the extraction of the camera methods from the library provided by the manufacturer. These methods allow a series of useful operations:

- Initialization of the camera: some operations, such the creation of the object itself (ueye.is\_InitCamera), the acquisition of the information of the camera (ueye.is\_GetCameraInfo), and the determination of the camera frame as “area of interest” (ueye.is\_AOI). If any of these doesn’t correctly perform, the camera cannot be used;
- Memory allocation: the operations needed to allocate the memory for the camera to save the image acquired, such as the allocation of the memory (ueye.is\_AllocImageMem) that determine the dimension of the data to be stored and the identifier of the memory used by the camera itself to find the assigned slot, the setting of the memory that links the memory information provided before to a memory slot in the pc (ueye.is\_SetImageMem), the setting of the color mode necessary for the correct storage of the image (ueye.is\_SetColorMode), and a method use for further inquire the memory (ueye.is\_InquireImageMem). These methods are strictly connected and sequential one to the other, and as for the initialization operations, the memory allocation is key to the correct working of the camera, and as such a fine monitoring should be implemented;
- Acquisition: once the preliminary operations are completed, the acquisition itself is managed by a single function (ueye.is\_FreezeVideo), that freeze the frame as the name suggest, fixing the data in the memory slot. Once the data is fixed, another method is called to save the current frame in the form of a three-dimensional array (ueye.get\_data);
- Exit: finally, before the end of the script the memory needs to be released (ueye.is\_FreeImageMem) and the camera object destroyed (ueye.is\_ExitCamera). The last operation is needed because the creation of a camera object univocally recognizes a camera via its identification number, thus making impossible to use that same camera again unless rebooting.

The initial code is used to test the acquisition.

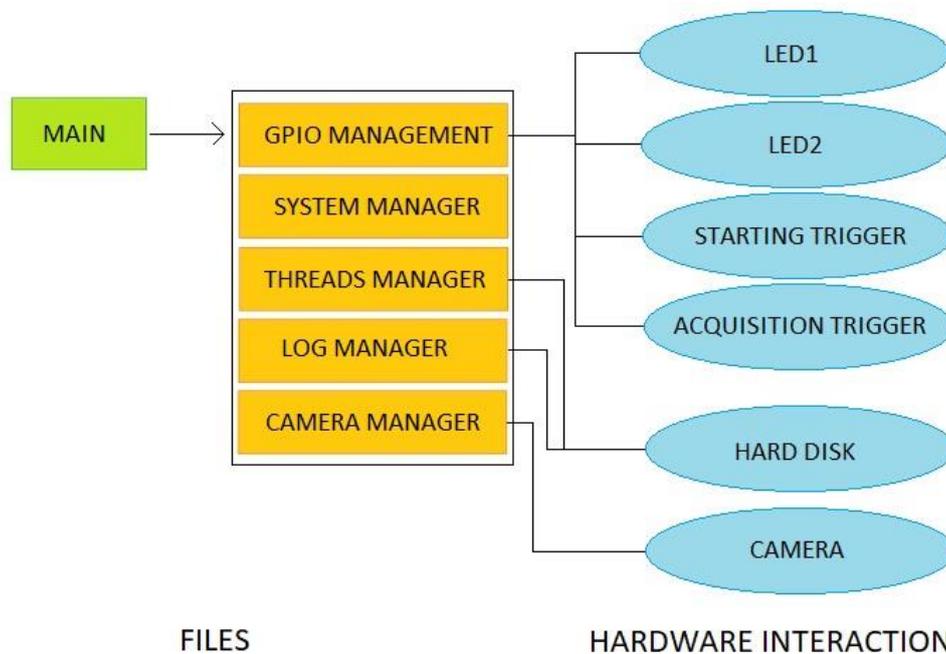
The acquisition results an array for each image. Due to memory limitation on the Raspberry Pi, the library used to convert the arrays into images is installed on a pc running Windows,

and a script is created to convert the images (fig. 5.1), plus the arrays are saved in an external memory connected to the Raspberry Pi in a folder created by the software with a timestamp for traceability. This script is presented here and with some modification will be used for all the versions of the software, so it will be illustrated at the end of the paragraph.

At first the acquisition is performed periodically, but this result in an asynchrony between the captures and the building of the layer, namely the acquisition is often performed in the middle of the layer, causing only a fraction of the objects to be completed and the melt pool to appear in the shot, causing a white area due to incandescence (fig. 5.2).

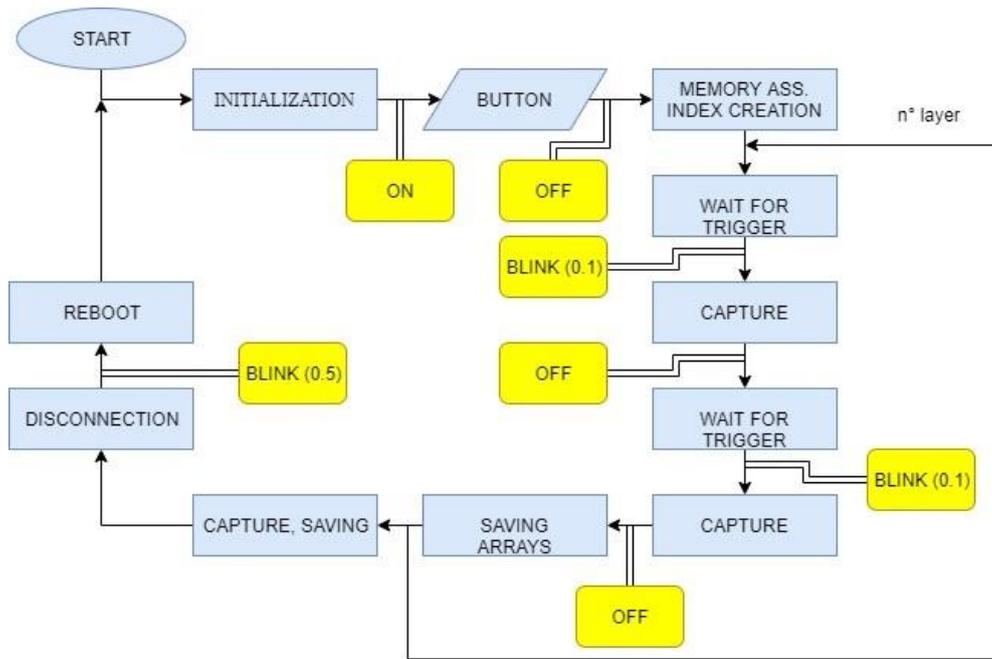
The target is to exploit an input from the machine to time the acquisitions. This is performed via hardware, and the software is adapted to read a GPIO input. A further improvement is achieved by switching to object-oriented programming (fig. 5.3).

The hardware implementation results in a calibration of the timing of the acquisition such that the system is able to capture images as desired (fig. 5.2).



(fig. 5.3) final code structure. On the right the links between files and hardware infrastructure.

An advanced version of the software (fig. 5.4) runs at booting and execute a reboot when finished, allowing the script to be launched without any user operation; a necessity, since the Raspberry Pi used wasn't equipped with peripheral devices.



(fig. 5.4) flow chart of the software. The script runs at startup and reboot at the end of the execution, creating a loop.

The starting of the acquisition cycle itself, however, is triggered by a button, since in the preliminary operations the opening of the door would cause multiple useless acquisitions (more later). Since no monitor is used, to inform the user of different states and signal any error in the operations, two LEDs are installed, and their functioning is explained in (tab. 5.1). In this version the number of layers is given as an information to the software, so that the correct number of acquisitions are performed. To add robustness and avoid the code remaining idle in case the number of layers is incorrect, a timeout is added to end the software if no acquisition is performed in ten minutes.

STATUS LED			
COMPLETE CYCLE	ONCE	ON	READY, WAIT FOR THE BUTTON
		OFF	WAIT FOR THE TRIGGER
	EVERY RECOATER'S PASSAGE	FAST BLINK	ACQUIRING
		OFF	ACQUIRED
	ONCE	SLOW BLINK (3 TIMES)	REBOOT

PRESS THE BUTTON WHEN THE YELLOW LIGHT IS ON TO START THE ACQUISITION ROUTINE

ALARM LED			
COMPLETE CYCLE	ONCE	SLOW BLINK (0.2)	ERROR IN MEMORY INITIALIZATION
		STEADY	WAIT FOR THE BUTTON TO REBOOT
	EVERY RECOATER'S PASSAGE	SLOW BLINK (0.5)	TIMEOUT EXPIRED, REBOOTING
		FAST BLINK (0.1)	ERROR IN ACQUISITION
		SLOW BLINK (0.2)	ERROR IN SAVING
	ONCE	STEADY	ERROR IN INDEX CREATION

PRESS THE BUTTON WHEN THE RED LIGHT IS STEADY ON. IN THIS CASE IT MEANS THERE'S AN ERROR IN THE INITIALIZATION OF THE CAMERA AT THE BEGINNING OF THE CYCLE

(tab. 5.1) two LEDs are implemented to signal the user, one for normal operations and one for errors.



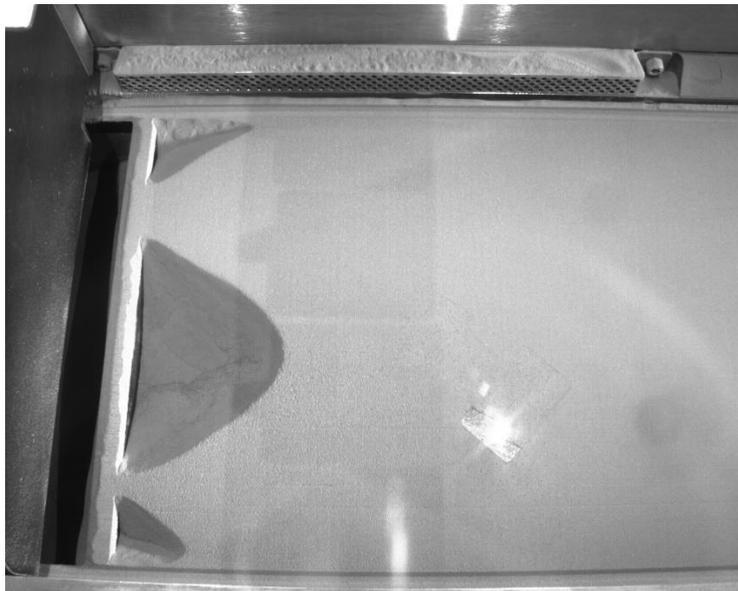
## Image analytics for in-situ defect detection in Additive Manufacturing

In this version, the creation of a LOG file is implemented, to allow monitoring and thus further improvements (fig. 5.6).

```
{"Cam_init":1,"Cam_info":1,"Cam_AOI":1,"Cam_memory":1,"Cam_queue":1,  
"Folder":"ok","Parameter":"err","Index":"ok","Export":[10,2200,3100],  
"Acquisition":[3,1015,3225],"Endtime":"2019:11:20_10:30"}
```

(fig.5.6) LOG example. The voice “parameters” is referred to the version in which the number of acquisitions is not dependent on a trigger but on a parameter given by the user. Alternatively, it is simply removed.

Finally, to improve performances some threads are created to execute operation that might slow down the response of the code, especially in the latest layers (fig. 5.7).



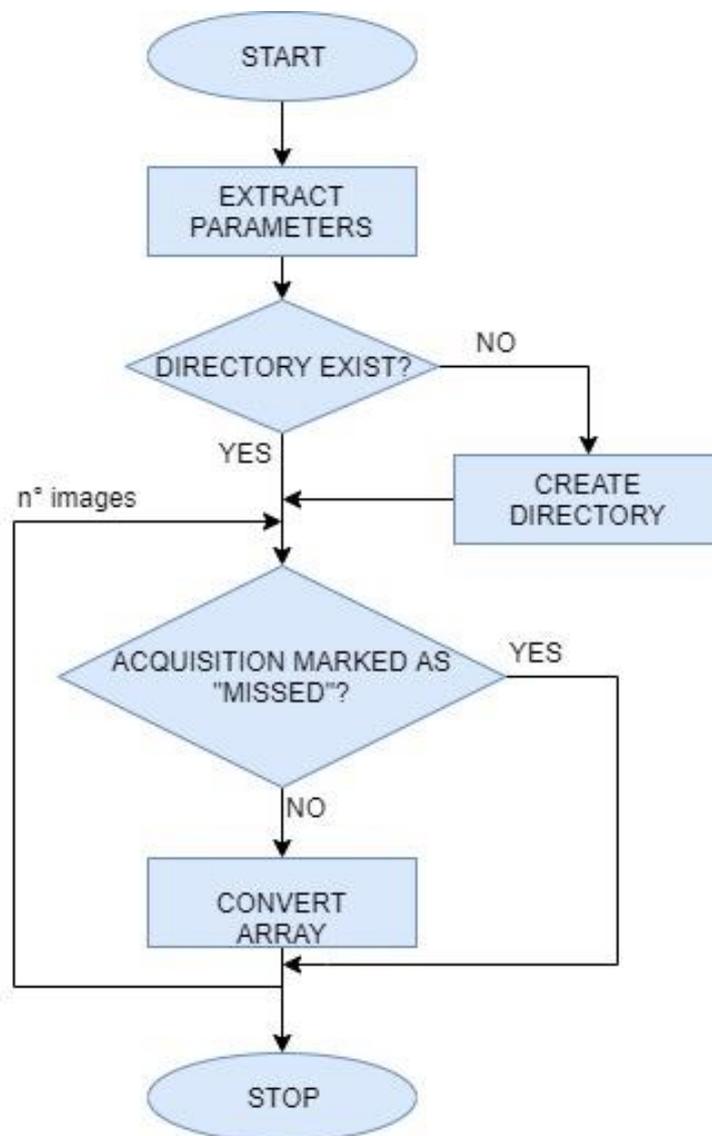
(fig. 5.7) latest layers often have reduced area, thus the interval between layers is reduced and it can outperform the acquisition script.

## Image analytics for in-situ defect detection in Additive Manufacturing

The conversion of the images is ultimately performed taking into account missed acquisition (noted in the LOG file as well as in a index created by the script), missed savings (errors in the storage) and all the information related to the frame, given by the camera method's and transmitted by the software.

To perform the acquisition the script must be placed in the same path as the acquisition folder and simply run. A destination folder with the same tag will be created by the script.

Nome	Ultima modifica	Tipo	Dimensione
20191015_1040_Capture	02/11/2019 18:16	Cartella di file	
uEyeDisplay.py	28/09/2019 12:45	Python File	2 KB



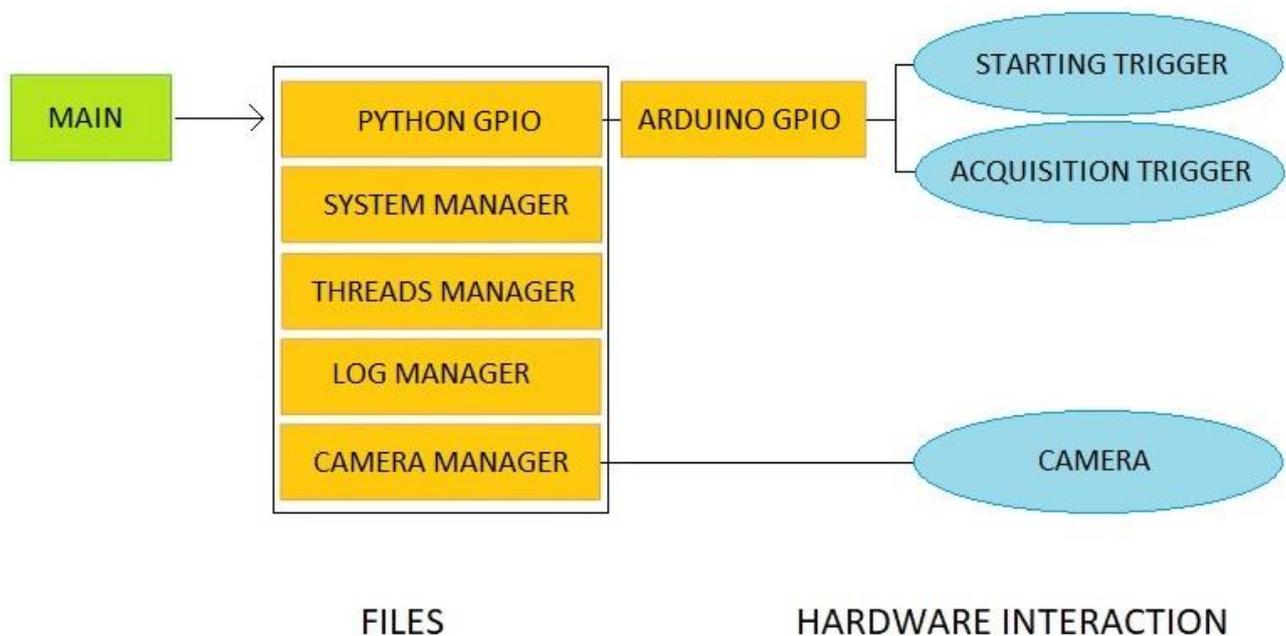
(fig. 5.1) above an example of how the folder needs to be filled before starting the script. Below the logic of the conversion.

## PC version:

Once the prototyping phase is completed, the mass acquisition of images can begin. The end of testing allows to select a more targeted infrastructure to achieve better results with better performances. The introduction of a pc running Linux and an Arduino UNO for IO management solves first of all memory issues, allowing to convert the images as they are being taken and storing them into local memory, skipping the storage of the raw array and saving in average 10GB for each acquisition, and preventing errors related to the external memory management, as now it isn't needed anymore.

Hardware improvements will be covered in the next chapter, although with the computer it was decided to install also a monitor and peripherals to interact with the software.

This addition provokes some modifications in the software: the possibility to use a monitor allowed the removal of any user-targeted hardware from the circuit, in particular the LEDs and the button, thus the scripts were modified accordingly. The implementation of Arduino UNO required a new code developed in Arduino IDE to communicate with the pc via serial port and handle the IO. Consequentially the script responsible for handling the IO in python was modified to interact with Arduino through serial communication (fig. 5.8).



(fig. 5.8) files structure of the PC version. Here the IO are directly handled by Arduino and then a serial communication allows data transfer to the pc.

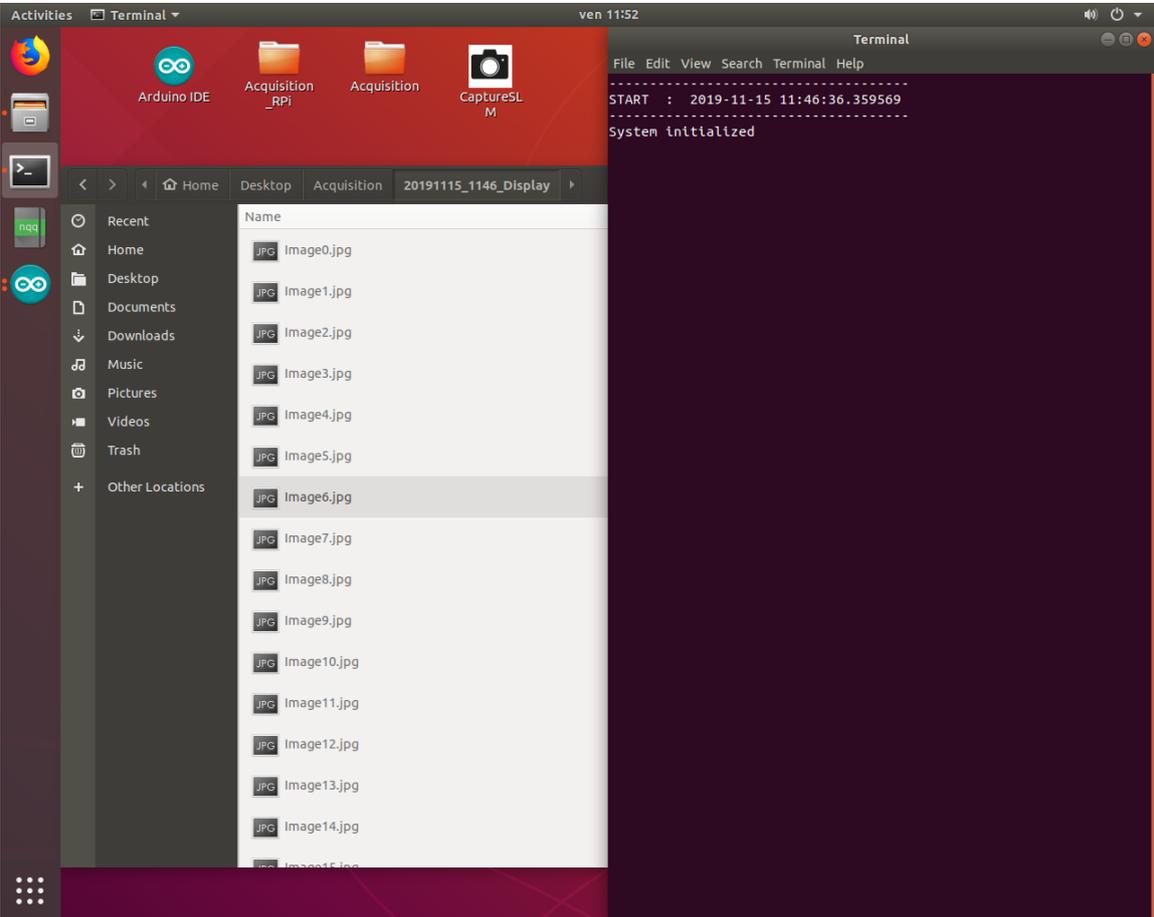
Finally, to start the cycle a desktop application was developed in order to maintain a user-friendly approach. An icon (fig. 5.9) must be pressed to start the software: this is linked to the python scripts stored in a folder and rendered available through the use of shebangs at the beginning of the code (fig. 5.10), and create a terminal window that represent the user interface, mainly with printed lines (fig. 5.11).



(fig. 5.9) Icon of the software, which is named “CaptureSLM”.

```
1 #!/usr/bin/env python
```

(fig. 5.10) shebang added at the beginning of each code to link the script to the



(fig. 5.11) On the right the terminal created at startup, on the left the folder automatically created with a timestamp where all the images are stored. In the background is visible the program icon, named “CaptureSLM”.

## 6. Hardware design

As aforementioned, hardware setup and software are developed in a codependent fashion, thus for each version of the acquisition a correspondent version of the hardware is developed. In the following the design process of the hardware setup will be explained, giving pros and cons of the solutions evaluated. At first the Raspberry Pi solution will be explained, taking into account the board characteristics, while later the version meant to be handled by Arduino UNO will be presented.

### Raspberry Pi version:

The desired acquisitions should frame complete layers and new ones. An observation of the machine working process identifies the recoater movement as a suitable trigger, thus the study of the electrical circuit identifies the driver of the recoater (fig. 6.1), a component that will be used to extract a signal.

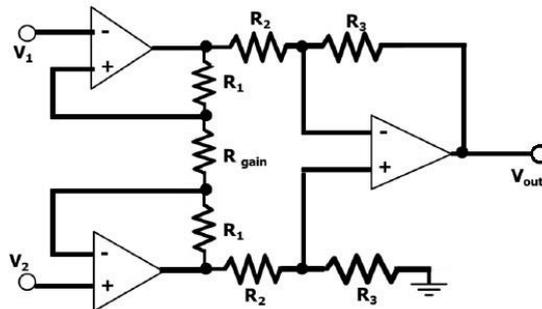


(fig. 6.1) Recoater's driver (ACOPOSmicro from B&R Industrial Automation, model). Visible on the front the LEDs signaling the status.

Three different solutions are considered:

The first solution is an instrumentational amplifier (fig. 6.2) installed on the cable that transmit the current to the recoater's electrical motor. This solution is widely tested for sensors, and it provide a current decoupling between the board and the driver, although it requires a physical connection to the machine and a comparatively large number of components. The design follows the relation below.

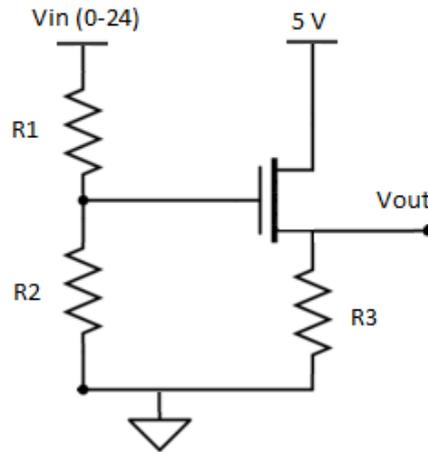
$$Gain = \left(1 + \frac{2 * R_1}{R_{gain}}\right) * \frac{R_3}{R_2}$$



(fig. 6.2) Instrumentational amplifier setup, the signal is taken between V0 and V1, with a load put in between.

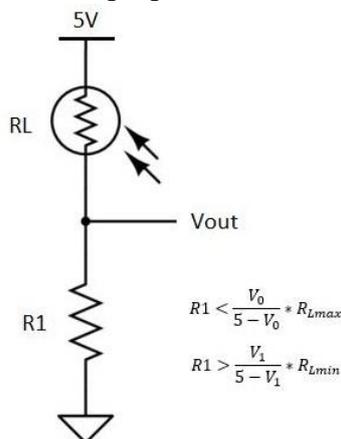
The second considered solution follows the same principle of the instrumentational amplifier, focusing on the simplification of the circuit. A simple MOSFET (fig. 6.3) is installed, that with the activation of the gate connects a voltage source (possibly the voltage output from the Raspberry) to the input pin. Although this solution is simpler than the instrumentational amplifier, it still requires a physical connection to the machine. The design follows the relations below.

$$R_1 + R_2 = \frac{24}{I_{max}} \qquad R_2 = R_1 * \frac{V_{th}}{(24 - V_{th})}$$



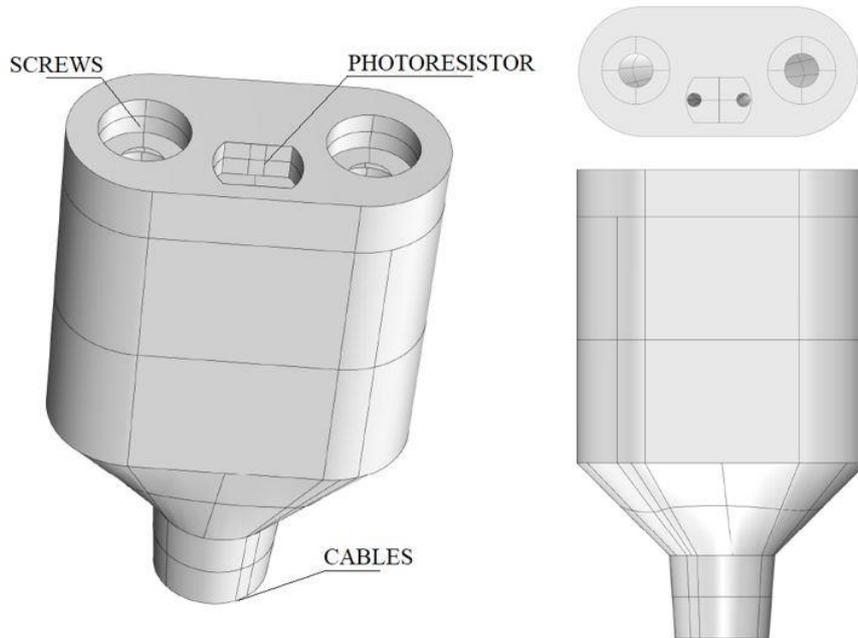
(fig. 6.3) MOSFET solution, R3 is chosen simply for current limitation.

Finally, a circuit is designed to attain information from the driver through the LED signaling the status of the recoater. This circuit exploits a photoresistor (fig. 6.4) to associate the resistance value with the LED being on or off. First of all, the LED is observed to be steady on when the recoater is moving, with different color (and therefore different intensity) depending on the direction of the recoater, while when the building chamber's door is open the led exhibit a blinking behavior. The photoresistor solution is very sensitive to different lighting conditions, thus it requires precise calibration and a protective structure. However, it also doesn't require any modification to the machine, being installed on the outside of the driver, saving possible issue with the manufacturer of the machine, avoiding the risk of damaging sensitive components, and simplifying the testing and the modifications to the circuit. For these reasons the selected setup is the latter of the three proposed.



(fig. 6.4) Photoresistor solution: V0 and V1 are the threshold values for Raspberry Pi pin.

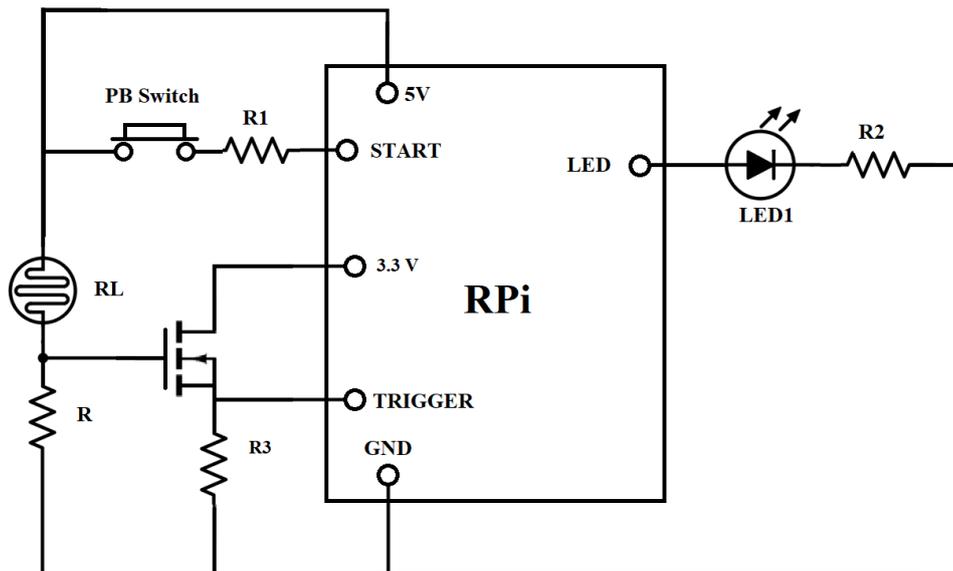
To guarantee enough lighting insulation a case is designed and manufactured with FDM technology available for prototyping (fig. 6.5). This case allows to expand the surface to adhere to the driver, thus allowing the usage of Velcro to guarantee that the photoresistor is in direct contact with the LED and sufficiently insulated from external light. In any case, since photoresistor's measurements are the basis for the sensor design, the key factor is that its range doesn't change in time.



(fig. 6.5) photoresistor case, designed in two pieces to be united after the photoresistor is installed inside.

Already from an early stage the software development requires other components, such as a button and a LED (later increased to two) to interact with the user in absence of peripheral devices. These components require current limitation, provided by some simply chosen resistances.

The first hardware setup is as follow (fig. 6.6):

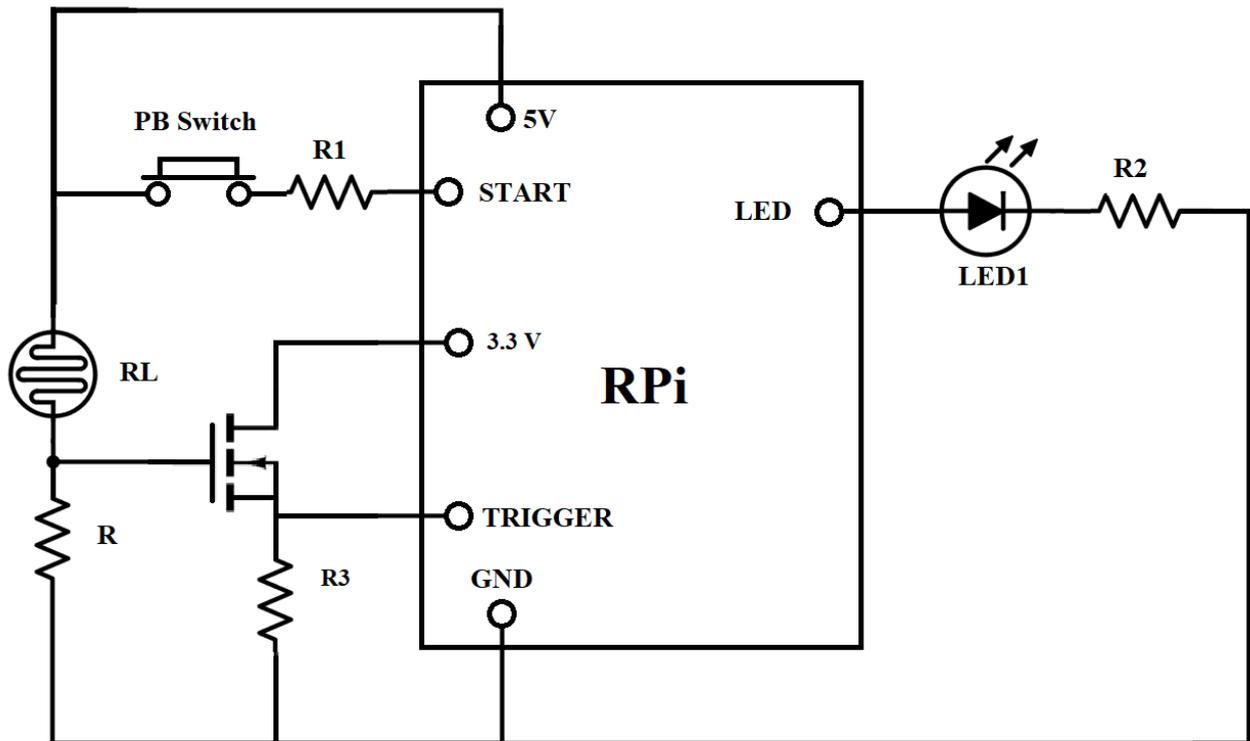


(fig. 6.6) In this setup the start of the acquisition routine was given by the button, thus only one photoresistor is present.

The designed circuit is then built and tested: results show an issue concerning the internal pull down resistor of the Raspberry Pi, that causes the voltage on the pin to be significantly lower, thus not being able to correctly deliver the trigger signal.

To fix this problem, the trigger circuit is modified as follow (fig. 6.7):

The voltage divider is maintained, and it's used to activate a MOSFET which will deliver the ON voltage to the GPIO pin. The resistor of the voltage divider is therefore designed such that the output voltage when the LED of the recoater's driver is on, is higher than the MOSFET's gate voltage. A resistor is added to limit the current through the MOSFET (tab 6.1).



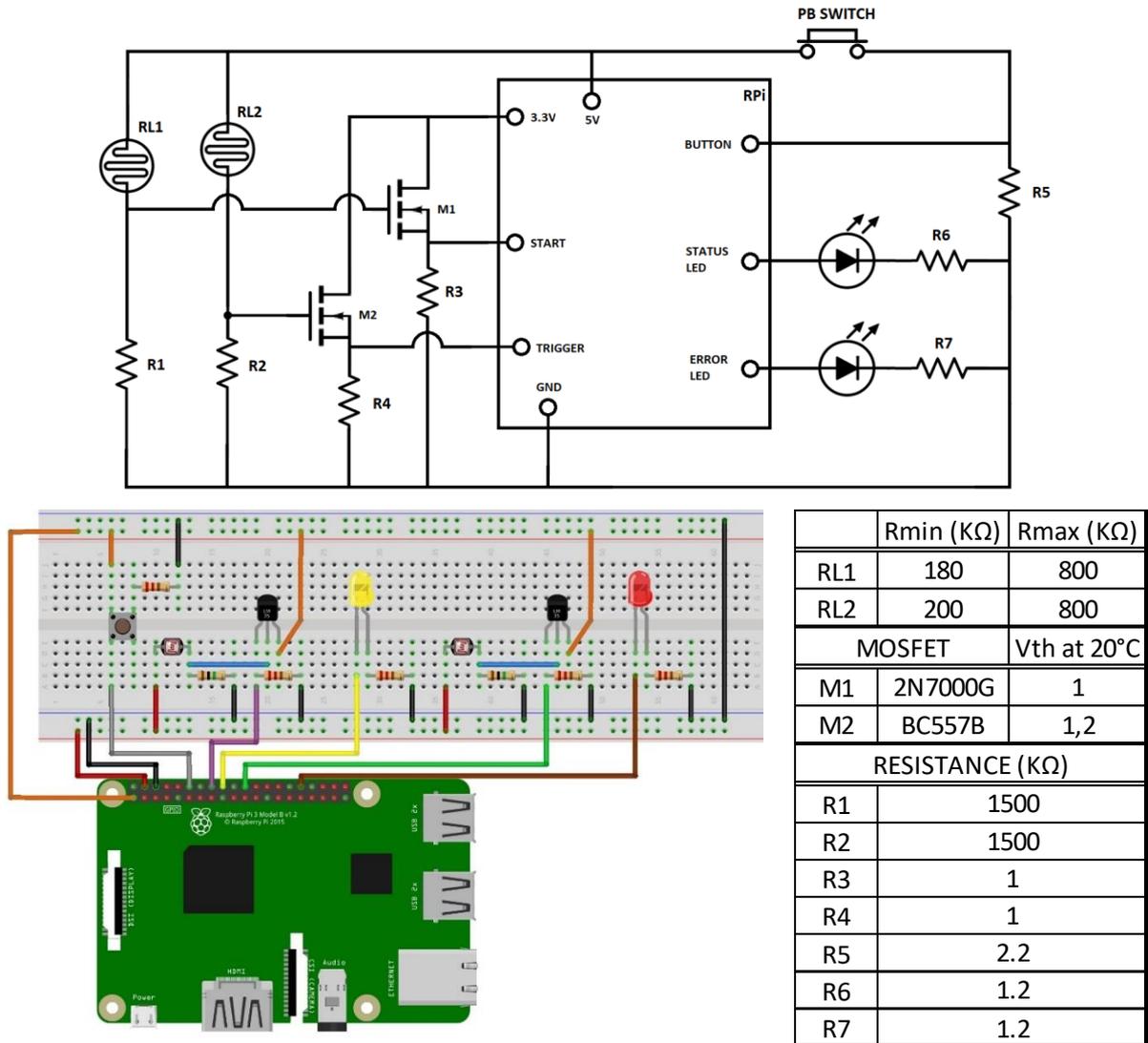
(fig. 6.7) the addition of the MOSFET allows greater precision in the measurement. Being the datasheet available to the public, only the model of the MOSFET will be reported.

The next improvement in the hardware setup follows the introduction of an input user-independent softwarewise.

An observation is conducted on the machine to identify a suitable event to translate into an input, and it is observed that the laser generation require the usage of certain infrastructures that only activate when the machine is ready to perform the manufacturing and stops when the building process end.

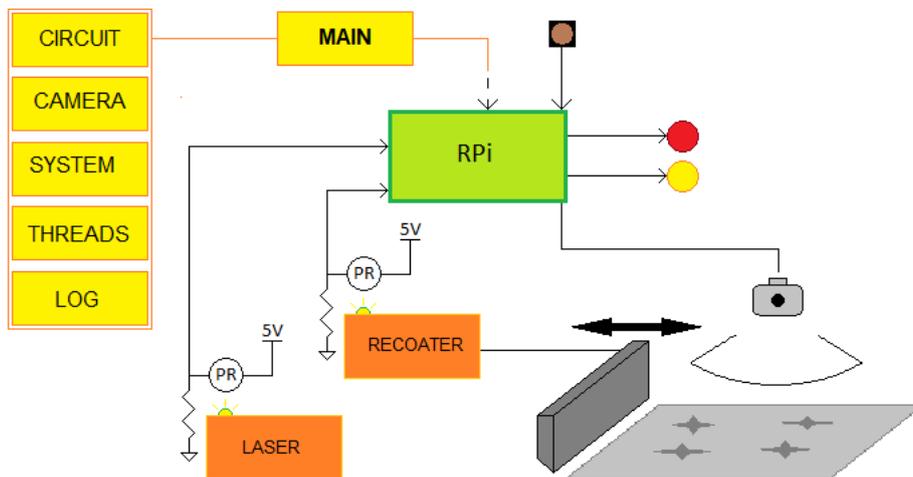
A study is then conducted again on the electrical circuit and the circuit regulating the laser generation is identified. The observation of the component during the working procedure show how a LED is used to signal if the machine is working. A sensor circuit identical to the one attached on the recoater's driver is then installed, and calibrated for the range of values that the photoresistor assume in the different states.

This addition leads to the final circuit for the Raspberry Pi version of the hardware (fig. 6.8).



(fig. 6.8) Final hardware circuit for Raspberry version. Schematic version on top and illustration on bottom left. (tab. 6.1) Component list on bottom right. Resistance values and MOSFET identification code with thresholds.

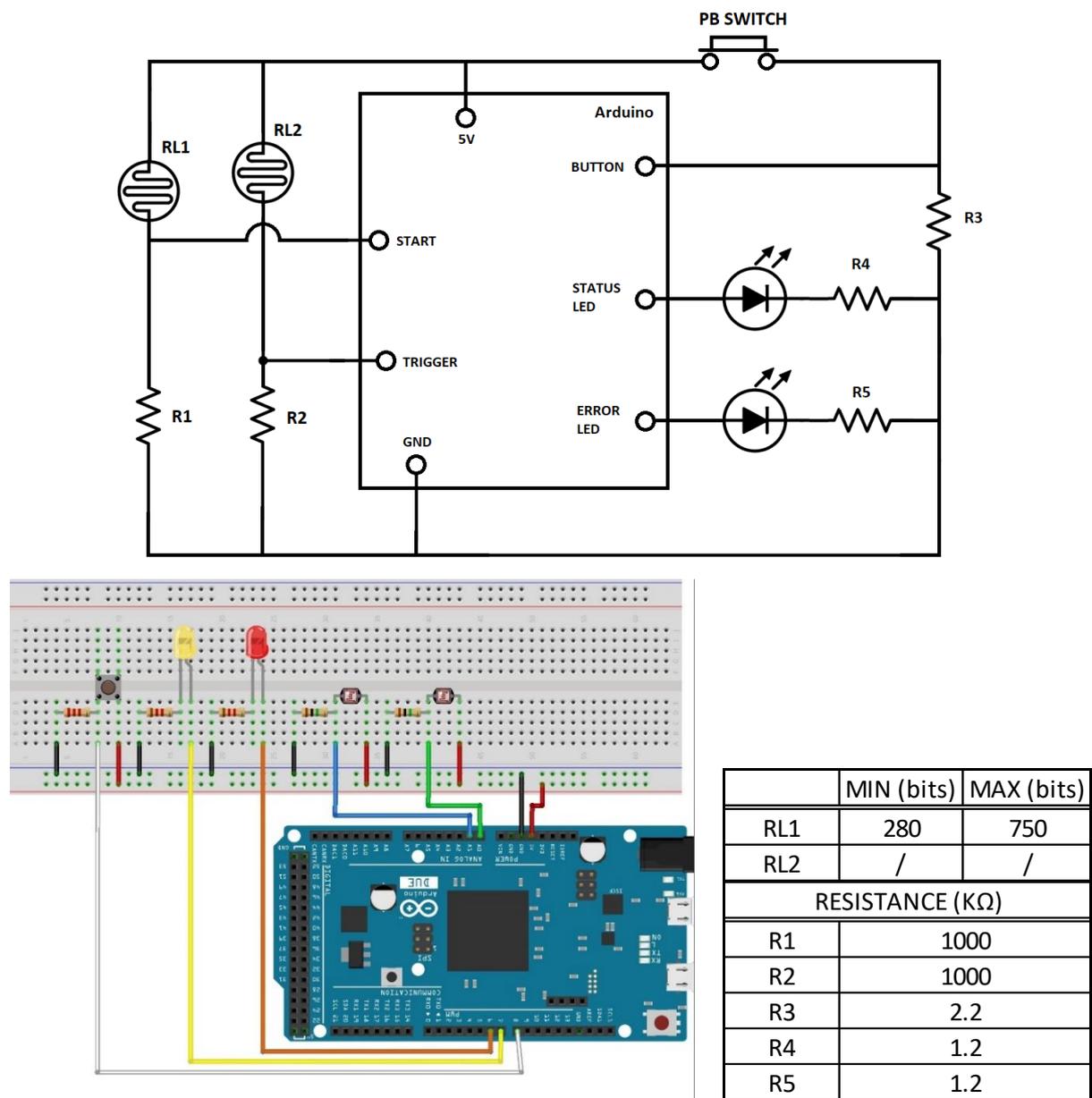
To conclude, an overview of the acquisition system is presented (fig. 6.9):



(fig. 6.9) General overview of the setup. In yellow the scripts structure.

Pc version:

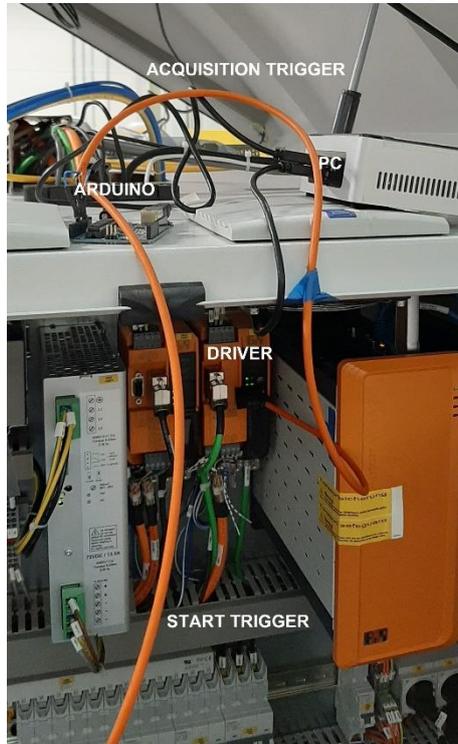
Following the switch to the software that runs on pc using an Arduino UNO board to handle the IO, the hardware faces some radical modifications: the main objective of introducing the Arduino UNO board is to go past the prototyping phase and in doing so improving precision of the acquisition. The first hardware setup in this configuration is a replica of the Raspberry Pi version, to quickly test the serial communication between Arduino and the pc without changing the scripts except the IO handler (fig. 6.10); it is worth noticing that the Arduino board, contrary to the Raspberry Pi, has analog inputs thus no precise calibration is required on the hardware end, and the MOSFET stage can be removed as the threshold is monitored via software.



(fig. 6.10) Arduino setup exploiting a bread board. Schematic circuit on top, illustration on bottom left. (tab. 6.2) Resistances value in pc configuration on bottom right. The photoresistor values refers to the value read by analog IN of Arduino board with the resistances R1 and R2. This value will be related to a chosen threshold.

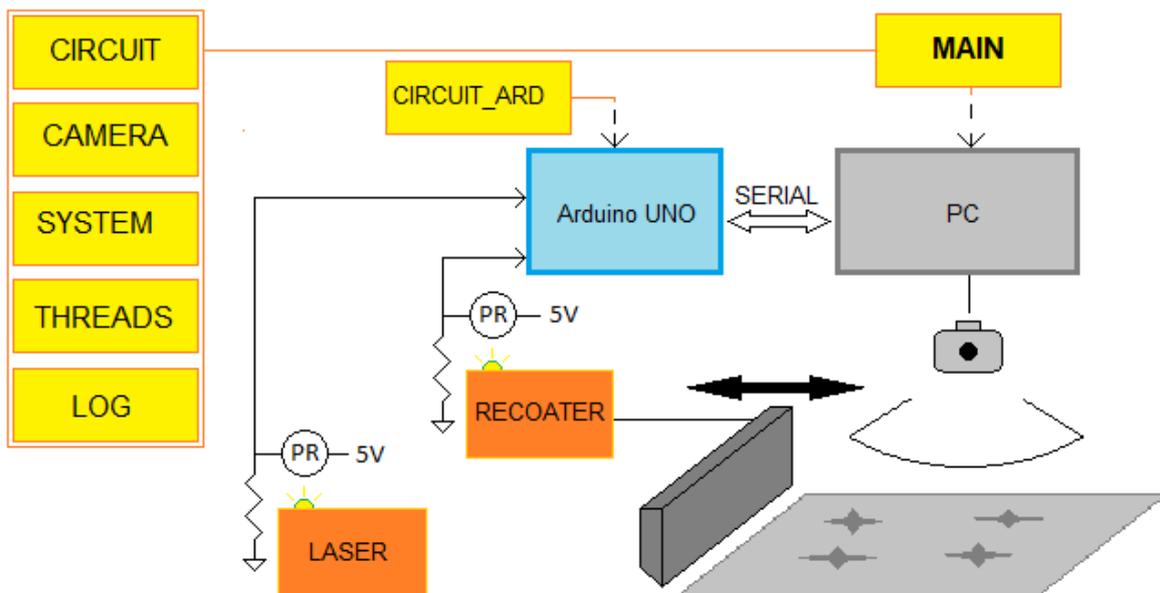
## Image analytics for in-situ defects detection in Additive Manufacturing

In the latest setup, instead, the LEDs and the button are removed as a monitor is implemented for user interface, thus it can be reduced to the sensors for start signal and acquisition trigger. To simplify the circuit tin soldering is used to connect the components inside the photoresistor case (fig. 6.5) using insulated cables that are connected directly to the board (fig. 6.11).



(fig. 6.11) cable installation on the electrical panel. Orange cables are the sensors going to the components, Arduino board and pc are posed on top of the machine.

Parallel to the previous version, an overview is presented (fig. 6.12):



(fig. 6.12) General overview of the setup. In yellow the scripts. The IO handling is managed by both Arduino and pc.

## 7. Image analysis

Once the image pool is created with the acquisition process, it is then analyzed to extract defect information. In this chapter, first the filters used are listed, describing the working principles, advantages and drawbacks, including the basic functions created ad hoc to be used by more functions. In the second paragraph the observed defects are described, focusing on the feature considered relevant. Next, the algorithms used for each feature extraction are described in a nutshell, and finally the validation process will be briefly explained.

### Filters:

Several functions are exploited in different ways to perform the analysis, mainly from the python library “scikit-image” (in short “skimage”), that provides a number of methods to filter the image:

- Canny edge detector: The Canny edge detection algorithm takes an image in input and return an image of the same dimensions with the edges of the original image. The algorithm works by using as edge operator the derivative of a Gaussian function, since it become easily computable for bidimensional images and its impulse response behave fairly similarly to an optimal step edge operator, and a hysteresis-based adaptive threshold to reduce the noise contamination [1];
- Sobel operator: the Sobel operator create an image in which edges are emphasized by multiply each pixel for a designed linear combination of its surroundings (3x3 matrix), resulting in an image in which each pixel represents the magnitude of the edge in the original image in that coordinate (fig. 7.1).

-1	0	+1
-2	0	+2
-1	0	+1

x filter

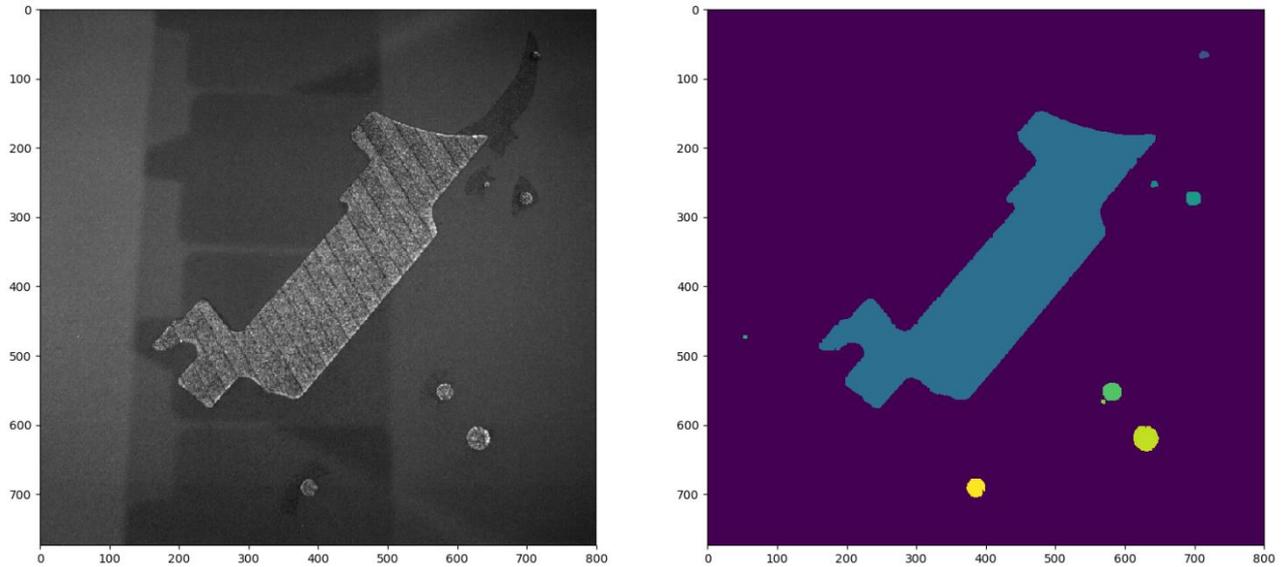
+1	+2	+1
0	0	0
-1	-2	-1

y filter

(fig. 7.1) Sobel operator, allow a pixel to assume a value that identify the characteristics of the surroundings.

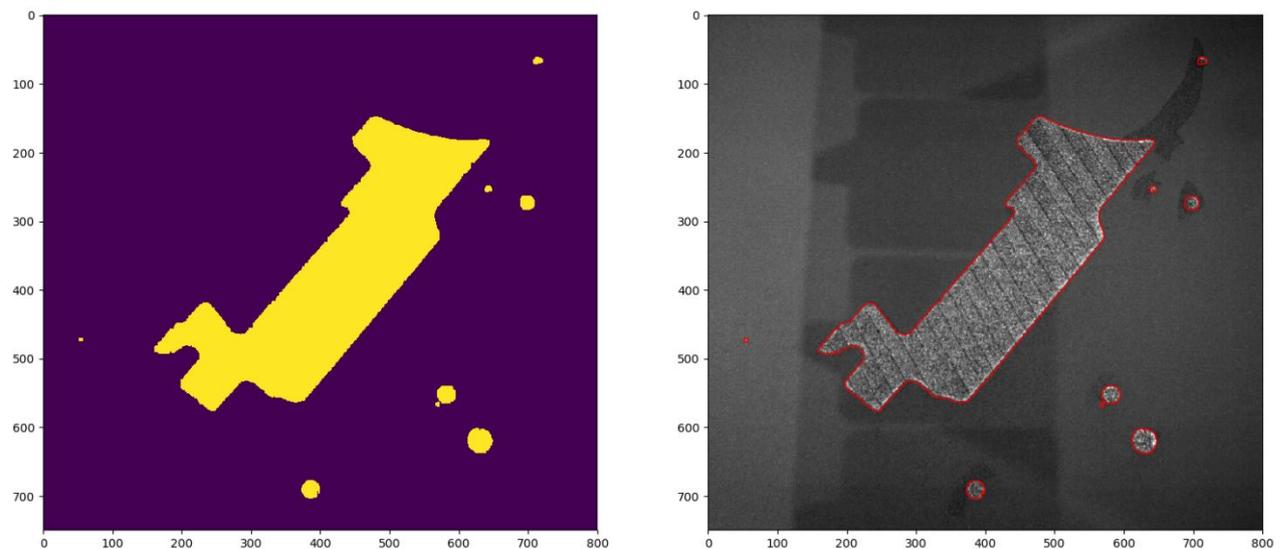
- Watershed function: the watershed function takes in input a mask with marked points and an image in which edges are depicted and combine the two to fill the edges starting from the marked points' coordinates. As the name suggests, in fact, the behavior is just like a water fountain, where the edges represent the limit of the fountain and the marked points the source of water [2];

- Label function: the label function works with binary like inputs, where the pixels with zero values are background and any other value is grouped together based on proximity and enumerated. The outputs are both a copy of the input array where non-background objects are substitute by the label value (fig. 7.2) and the number of distinct objects identified [3];



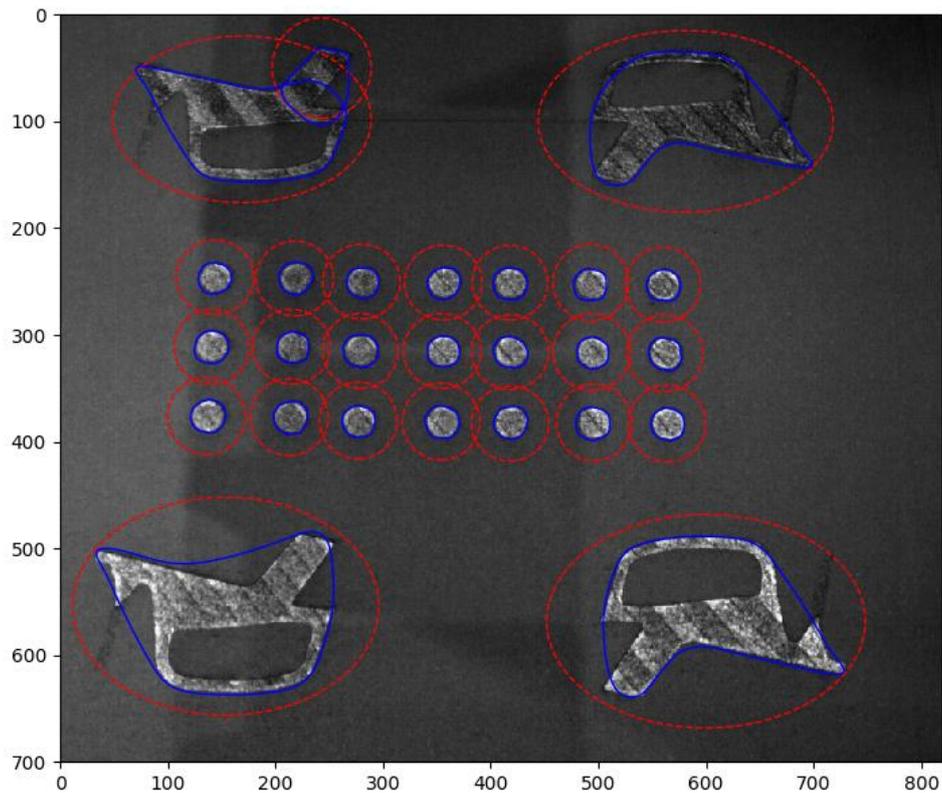
(fig. 7.2) the identified objects are numbered. Graphically, a different color is assigned to each number, the background being “0”.

- Probabilistic Hough transform: this function, described in [4] in the chapter “State of the art”, allows to identify lines inside the target image, provided a reference length. The result will be then filtered and used in defect detection. Although computationally performing, the use of this function involves a probabilistic approach, meaning there is the necessity to guarantee robustness through other parameters;
- Contour finding: a function called find\_contours allows to extract a line in the form of an array from a region in the image, where the line is the contour of the region of interest, provided the value that the function should consider threshold (fig. 7.3);



(fig. 7.3) on the left the segmented image, on the right the contours in red, applied to the original image.

- Active contour: this function implements the snake operator described in [5] in the chapter “State of the art” takes in input an image and a close line in form of array and remodel the line to find an object, returning the array correspondent to the contour of the object (fig. 7.3). Different parameters are to be set to influence the line characteristics, such as the speed at which the line contracts, the smoothness, the attraction to brightness, or the attraction to edge of the object;



(fig. 7.4) An initial contouring containing the target object (red) is used to generate the snake (blu). The tuning of the parameters allows for a stricter identification of the borders.

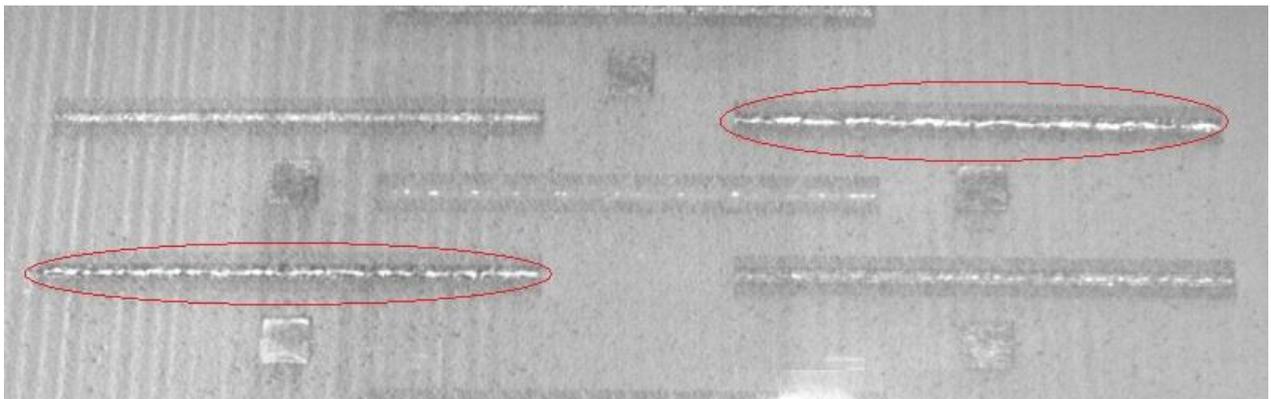
- Polygon: the polygon function generates a polygon given an array of points that constitutes the correspondent perimeter. Different geometrical operations are then possible, from area computing to extraction of specific coordinate (for example the farther coordinate belonging to the object, or the center)

## Defect identification:

The list of defects hereby described comes from an empirical observation of irregularities a pool of images initially acquired, without a-priori knowledge of the precise effect that these irregularity might have on the finite piece (except for some study performed on the spattering generation, described in the chapter “State of the art”).

Due to this condition the analysis code will be structured in a modular fashion, so that if any new observed irregularity arise it is possible, with minimum effort, to comprehend it in the image analysis algorithm.

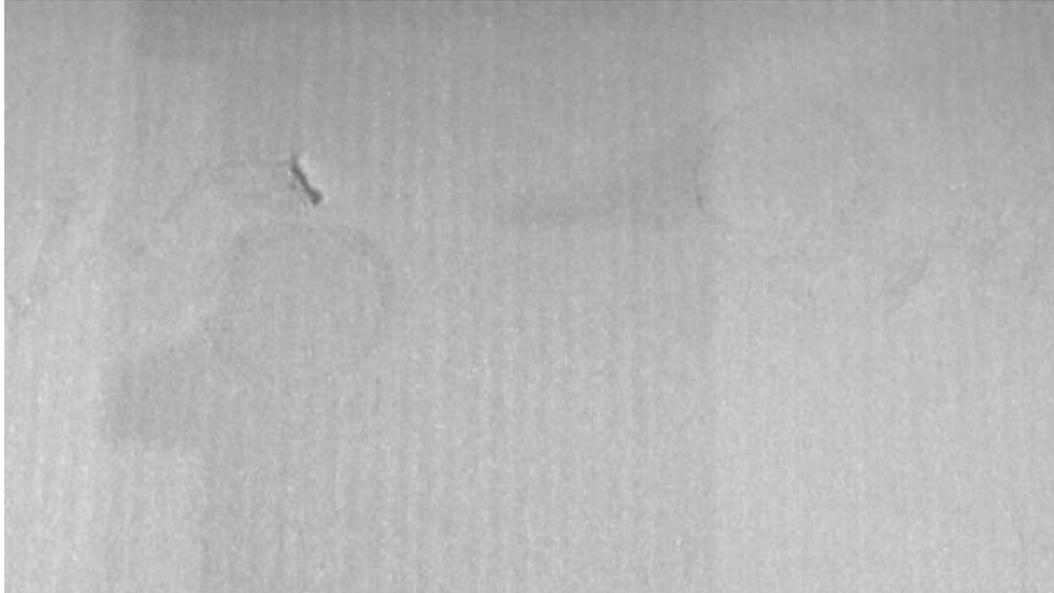
- Incandescence: The first observed defect is an incandescence appearing in the complete layer, resulting from an energy density to high that result in the inability by the melt pool to correctly cool down. In the observed cases this defect appears more easily in the center of the object or where the laser last passage was, as these are the areas that better retain heat. In the captured image the incandescence appears as an area with high intensity with respect to the neighborhood (fig. 7.5);



(fig. 7.5) example of incandescence.

## Image analytics for in-situ defect detection in Additive Manufacturing

- Holes: This kind of defect consist in a lack of metallic powder in the next layer, resulting in the appearance of dark small areas (fig. 7.6), since the powder doesn't cover up the melted layer below (darker than the powder). The origin of this defect might be the lack of powder due to the regulation of the dosing factor, as well as irregularities in the previous layer in terms of geometry of the manufactured artifact that result in the powder distributing unevenly. It can be corrected by adjusting the amount of powder supplied layer by layer via a manual regulation;



(fig. 7.6) example of a hole in the powder bed.

- Spattering: as aforementioned spattering defects are well described in the chapter “State of the art” and consist of droplet of melted metal ejected from the melt pool and landed in the surroundings. It appears as a spread of dark imperfection with small area, outside the object contour (fig. 7.7);



(fig. 7.7) the droplets constituting the spattering are visible as dark points in the bed.

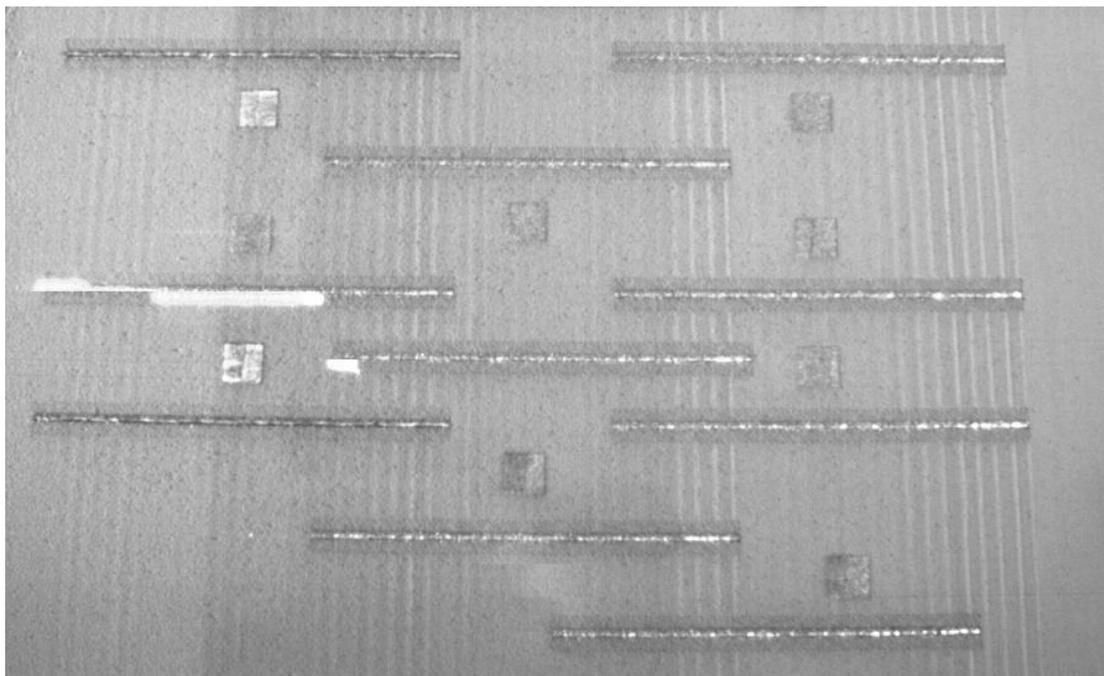
## Image analytics for in-situ defects detection in Additive Manufacturing

- Horizontal: defect labelled as “horizontal” consist in dark lines in the powder bed, outside the object contours (fig. 7.8). The origin of these defects is debatable, as no significant literature has been found on it. One hypothesis is that geometrical imperfection in the piece being build result in an incorrect behavior of the recoater that doesn't spread the powder correctly. It is also possible that the cause are imperfection in the powder itself, as one prominent feature of the SLM additive manufacturing process is the possibility to re-use powder, thus spread imperfection in the powder could be supplied to a new manufacturing job (for example droplet of melted powder);



(fig. 7.8) the horizontal lines are visible on the left.

- Waves: these defects, as the name suggests, appears as undulation in the powder in the direction of movement of the recoater. In the frame these appear as vertical alternation of light and dark stripes (fig. 7.9). The origin of this defect is geometrical imperfection in the manufactured objects, which cause the recoater to impact the solid artifact, resulting in an oscillation along the z-axis and thus the spread of the powder with the described pattern.

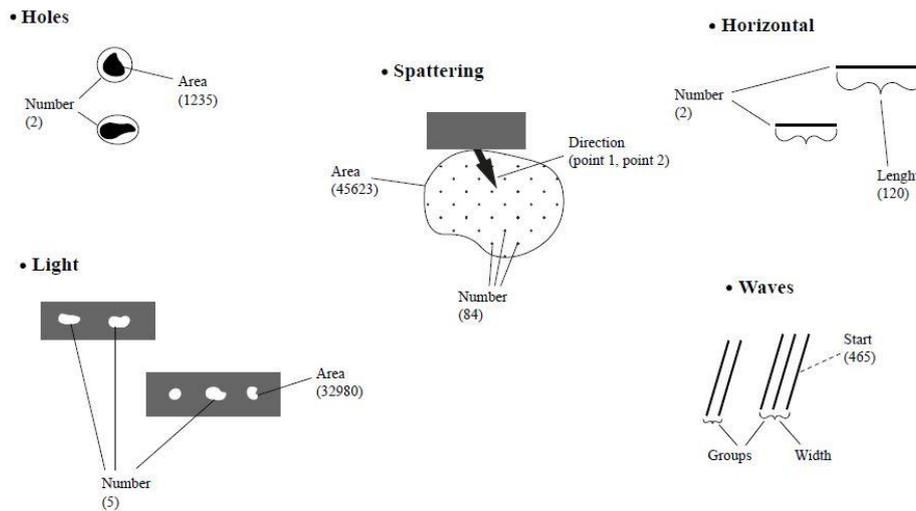


(fig. 7.9) the visible waves start upon contact with manufactured pieces.

## Feature extraction:

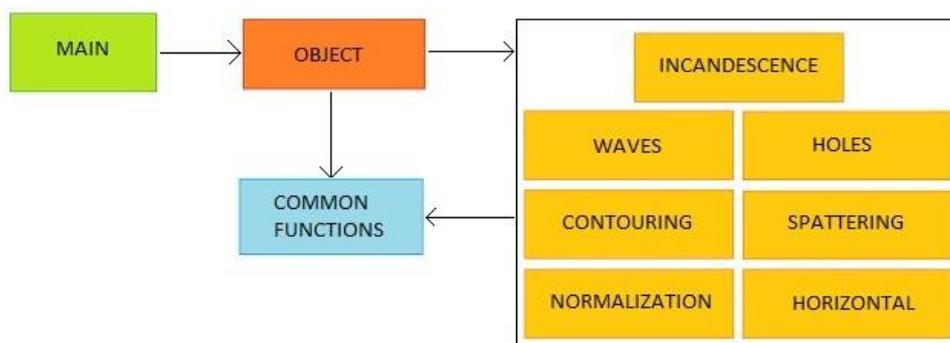
The feature extraction algorithms are described in a schematic way, mentioning the used filters and functions, and explaining the issue to be solved that required certain solutions. Initially an overview of the main algorithm will be given, describing the organization of the files and the logic of the main file. Then the single algorithms will be explained as above mentioned, and for each an example of the result will be given. Finally, the validation of the algorithm will be explained, and the results presented.

Relevant features are probed (fig. 7.10), mainly indications of quantity (such as number, length and area covered), and position, to allow future correlation of defect to one another.



(fig. 7.10) other than the features listed in figure, the position is noted, namely the defects are assigned to a sector of the image. The division in sectors used was a 3x4 grill. The number of sectors can be changed to increase precision.

The structure of the analysis algorithm consists of a main python script that handle the data flux, feeding the images to the different files and collecting the resulting features to build the output. The main tool of the scripts is the class handler that stores the parameters for defect identification and practically calls the methods from the other files. Alongside these two actors there is a different file for each operation performed on the images, namely seven: holes extraction, horizontal extraction, waves extraction, incandescence extraction, spattering extraction, normalization of the image, and contouring of the image. Finally, a file containing useful functions is created to be accessed by both the class handler and the single functions (fig. 7.11).



(fig. 7.11) each box is a distinct file, and the arrows are calls of the function.

### *Main*

The main file takes the image pool and the process information as input. Initially the code extracts, if present, the last image of the job to be used for normalization of the images (as it is a capture performed once the job is concluded and thus only contain powder, allowing to account for illumination conditions). Its first operation on the images is to check that the image number isn't registered in the "missed" list from the acquisition algorithm (see chapter "Acquisition script"), and if the image is present and the background image was extracted, it perform a normalization. Then, since the number of the image identifies if it is a capture of the complete layer or a capture of the newly posed powder, and different defect are recognizable in either one of these two cases, the current image is fed to a different branch depending on the number in its name. In the branch containing the complete layer images, some defects need to be fed the contours of the manufactured artifacts, thus the contouring is executed and passed as input to the functions that require it, and the defect features are stored in a dictionary. At the end of the cycle the dictionary is saved in an external text file, with the process parameters.

### *Detective*

The object handler is a class containing a method for each major operation that might be needed, although the operations themselves aren't in it. Some of the method contained are for example the feature extraction for the defects, that are execute each in its own file.

The object handler, in this case, calls the function to perform the extraction, then rearrange the information in a suitable format to be stored in a json file. More importantly, this is where the parameters are assigned and passed to the single files, so that any tuning needs to be done inside the object handler class (for instance the tolerance for defect dimension, or the thresholds needed).

Other than the feature extraction, the object handler contains also the normalization functions and the contouring, that return the image given in input after performing operations on it. In these cases, the object handler only serves as a link between the main loading the image pool and the functions.

### *Normalization*

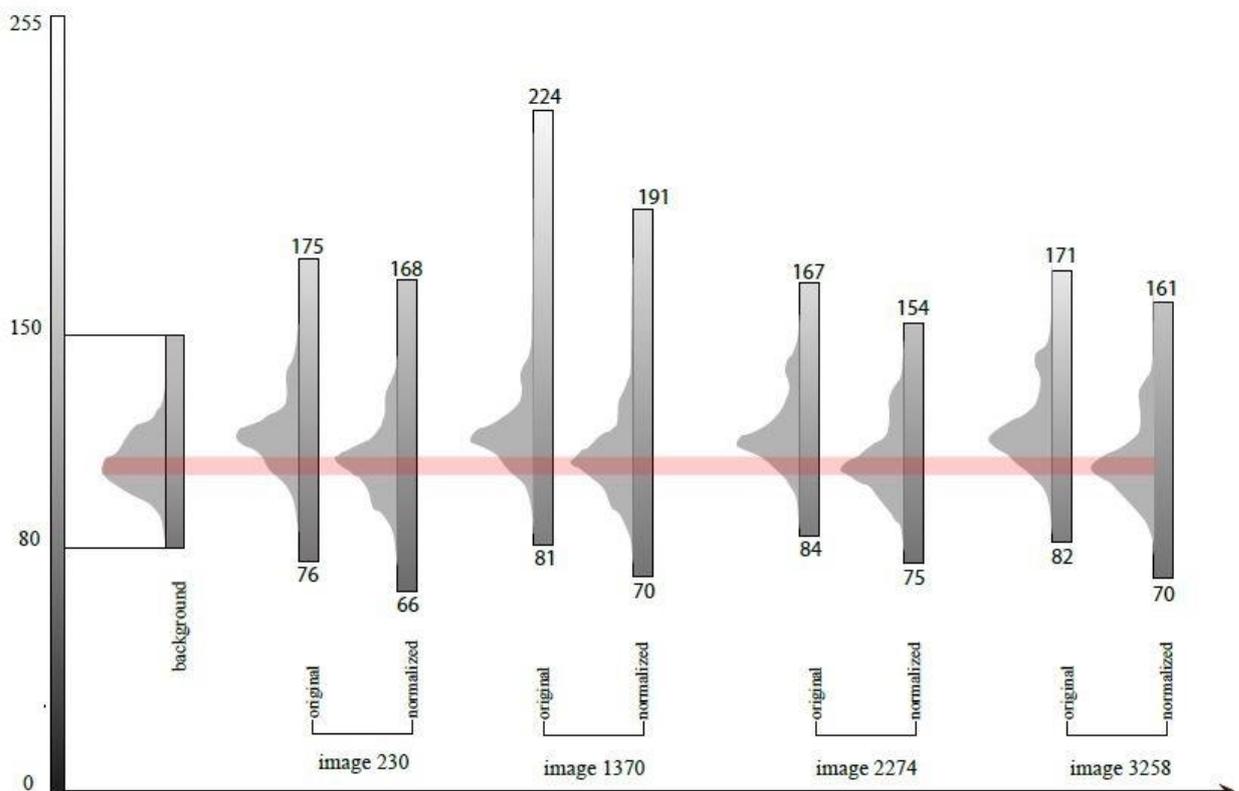
The camera used automatically adjust the luminosity according to the brightness conditions of the environment, resulting in different intensity level depending on the acquisition. In particular, if the new layer acquisition is a bit too slow and the laser start the action when the frame is being captured, the overall intensity experiences a spike, while the complete layer acquisition happens when the recoater has just passed in the frame. The presence of such a huge darker shape in the frame makes the camera react by lowering the overall brightness, resulting in a lower intensity in the acquisition. However, the tuning of the parameters starts from the assumption that is possible to identify values valid for the whole image pool (to avoid complicated dynamic assignments), thus a function is necessary to normalize the images, described below.

The reference image is the extra capture after the end of the acquisition routine, since it is guaranteed to have only powder in the frame, thus providing a reliable reference to identify the illumination conditions. This image (from here on called "background") is divided in levels of intensity with a resolution assigned by the user (for example, if the designed value is "5", each pixel between a multiple of 5 and the next multiple belongs to the corresponding level), then the mean value of all the pixels belonging to a certain level is computed, and that

value overwrite all the pixel considered to compute it (a similar process was to round every pixel to the closer multiple of a certain resolution level, but for large range values this would have result in a significant shift from the original value). The image resulting from this operation is a mask containing reference values that allows to distinguish different region in the frame. The creation of the mask is time consuming, but it needs to be executed only once at the beginning of the cycle.

The target image is segmented to identify the objects, not necessarily in a precise way but just to exclude them from the intensity computation, as the different consistency of the material means also a different intensity. The image is then divided in groups following the coordinates of the mask's levels, and for each group the mean value is computed (excluding the objects), a the difference between this value and the correspondent mask value is subtracted from the pixels belonging to the group in the target image (including the objects). After this operation is performed the normalization is completed and the area containing only powder in the target image assume now the same value as their correspondent in the background image, while the difference between the powder and the objects in term of intensity is maintained untouched (fig. 7.12).

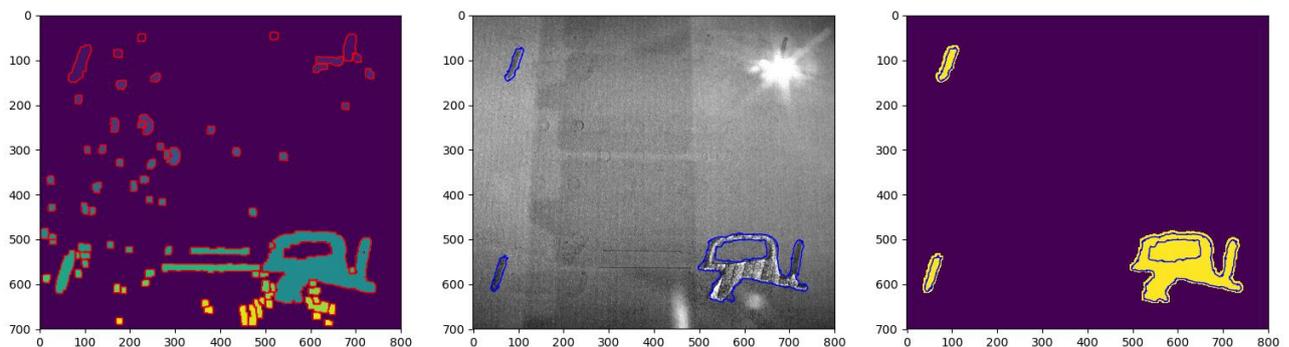
The main advantage of this method with respect to the simple histogram equalization of the image and the background is the ability to exclude the objects from the computation, as they would significantly influence the outcome, and still maintain the difference between objects and background unchanged.



(fig. 7.12) while the range remain fairly higher due to object presence, the histograms of the normalized images are shifted to match the background.

### *ActiveContouring*

The aim of this function is to create a mask in which the objects are identified using a snake-based active contouring (as referenced in the chapter “State of the art”). The first step in the contouring is the creation of the first snake that will be later brought to completion by the function in the python library “scikit-image”. The target image is initially segmented with a fairly low threshold, to ensure the identification of all the objects. The segmented image is then labeled (the same integer value is assigned to adjacent pixels recognized as object) and then “expanded” assigning to the pixels surrounding the object the same value as the object of interest. The expansion also take care that there is no merging of distinct objects, allowing to contour the expanded object, creating lines that surely contains the objects and that serve as the first snakes (fig. 7.13). The active contour function is then implemented on a copy of the image filtered through a Sobel operator for noise removal, tuning the parameters to be as robust as possible, and the resulting lines are used to create a mask where the pixel value is “0” if it belongs to the background and “1” if it is inside one of the snake.



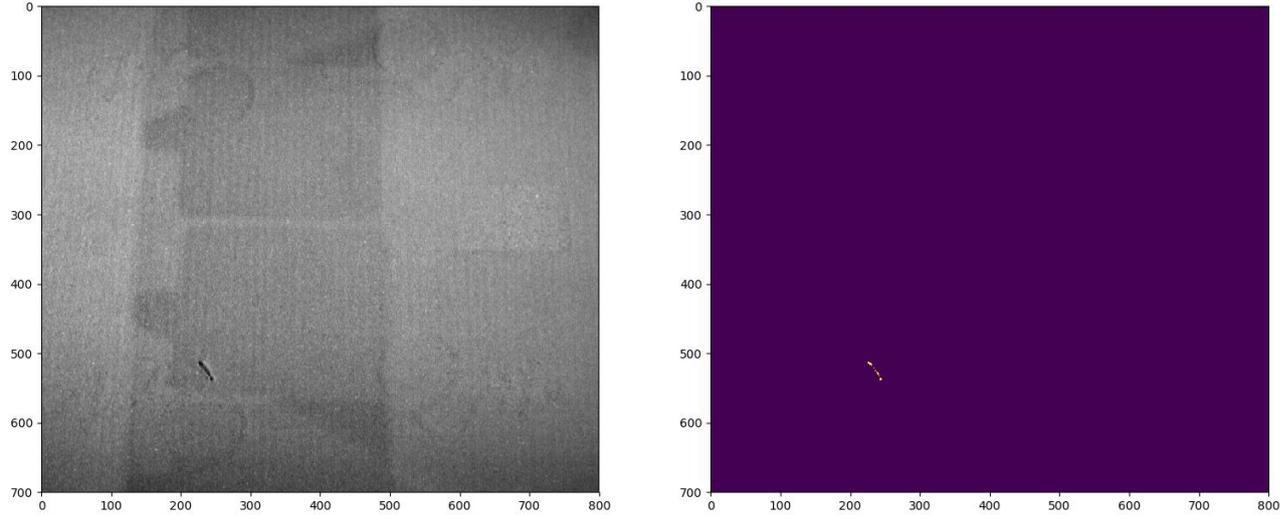
(fig. 7.13) the segmented image is expanded and contoured (on the left in red), and from that a snake is implemented (on the right in blue) filtering for dimension. The snakes will be then filled to create the mask (on the right).

### *Contouring*

This function aim at performing a segmentation of the image to identify the objects and then filter the segmented image by dimensional characteristics that exclusively belong to defects (such as very small area). The main advantage of this implementation with respect to the active contouring function is the time performances, but the withdraw for this is a significant loss in precision, as the object is no longer identify by a targeted function.

### *Holes*

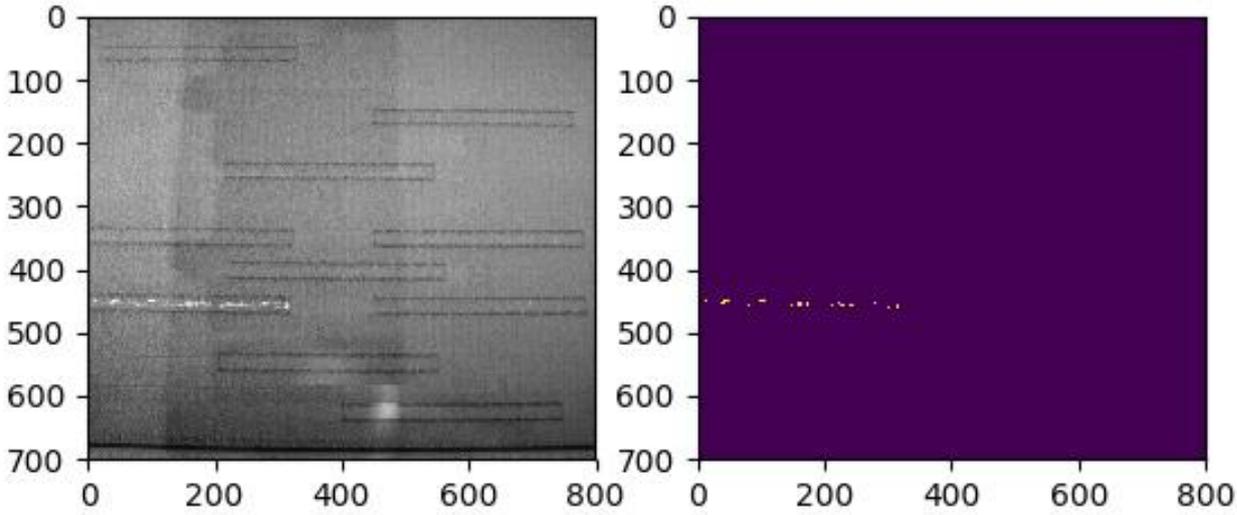
The target of this function is the extraction of the relevant features regarding the defects categorized as holes. At first the image is filtered to highlight discontinuities by means of a gaussian filter and a Sobel operator, then a mask is created containing the point of the image below a certain threshold given as a parameter, and the watershed function described above is used to identify object with values below the threshold. The result is labeled to easily count the number of distinct defects and compute the total area and filtered to remove the objects outside the range of feasible area values given as parameters. The latter is done to add robustness, accounting for a situation in which the reactor is in the frame during the capture, and also during the final layers when a heavy partitioning of the powder is required and the lack of powder in the building site form a huge shadow (fig. 5.7). Finally, a function is used to identify the center of each object, allowing to recognize the position (fig. 7.14).



(fig. 7.14) Holes identification.

*Incandescence*

The target of this function is the extraction of the features regarding the defect categorized as incandescence, consisting of part of the object that doesn't correctly cool down after the manufacturing. The appearance of this defect is symmetrical to the one of the holes defects, allowing the implementation of the same procedure but marking the bright spot in the image. In this case it's also implemented the contouring to add robustness, as to identify only the point inside the object where the incandescence might actually happen (fig. 7.15).

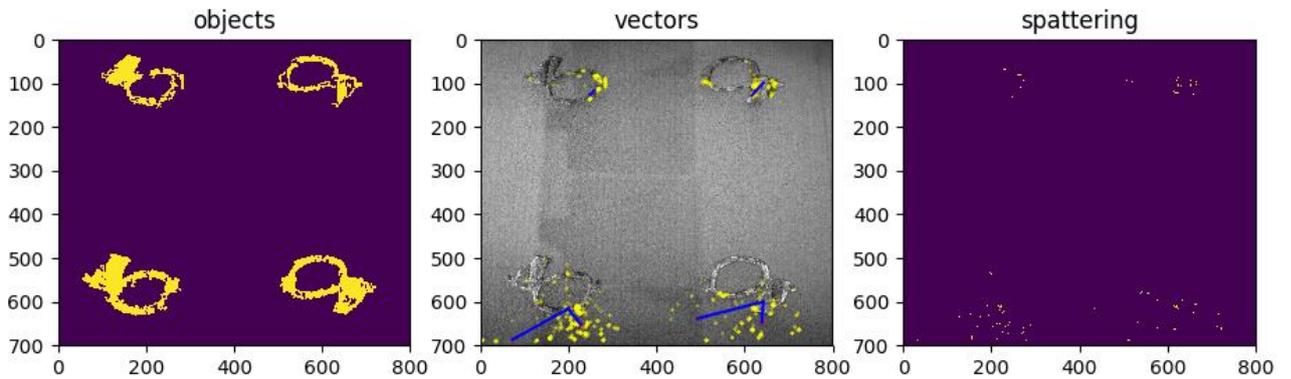


(fig. 7.15) Incandescence identification.

*Spattering*

The function probing the presence of spattering exploit both the reduced dimension of the spatter, and the darker appearance caused by the difference in composition with respect to the powder. The identification of the defect is performed at first by segmenting the image with a low threshold in order to identify even the smaller particles, then the segmentation is filtered to remove the objects (given as input as a product of contouring) and the particles with intensity higher than a defined threshold (since the segmentation also identify areas with intensity higher than the surroundings. With the resulting information the number and total

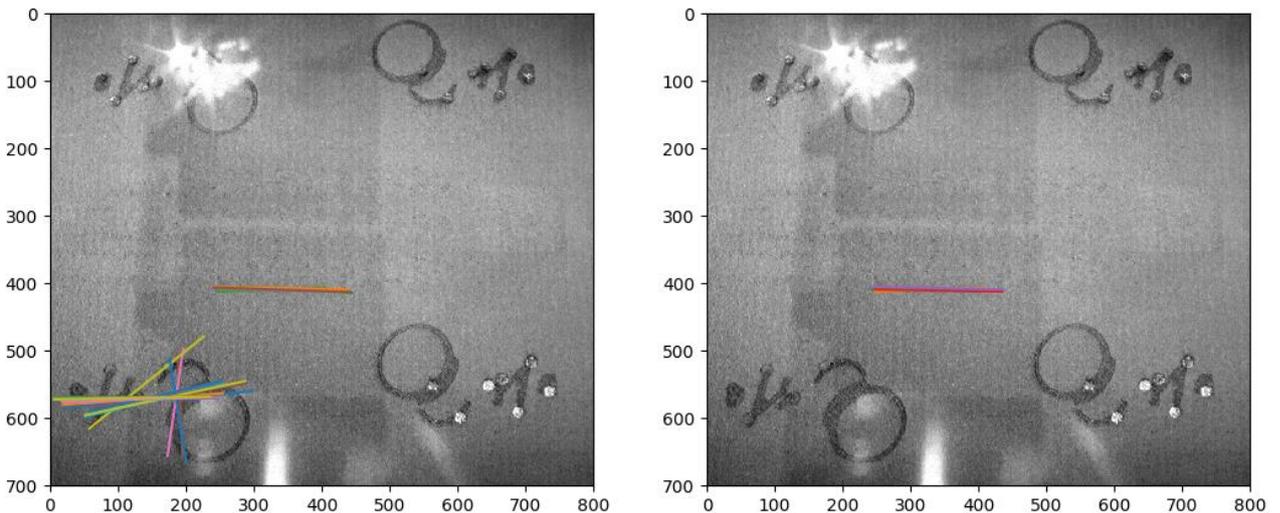
area are computed, as well as the mean area, and finally a K-means clustering is used to associate the spattering area to the objects identified, in order to probe the existence of a preferential direction of spattering generation (fig. 7.16).



(fig. 7.16) Spattering identification. In the middle, the vectors connecting the center of the nearest object and the center of each cluster of points are highlighted in blue.

### Horizontal

The horizontal defects identification algorithm exploits the progressive probabilistic Hough transform presented in the “Filters” paragraph, applying the transform to a segmentation of the target image to identify the lines present. The result is then filtered to remove the lines not horizontal, exploiting the signature characteristic of this class of defects, and to remove the lines identified inside the objects. For robustness, the lines identification is repeated with a range of target length, possibly causing the identification of the same line multiple times. To account this dynamic a function is implemented to merge together line that identify the same defect, based on the coordinates of the two end points (fig. 7.17).

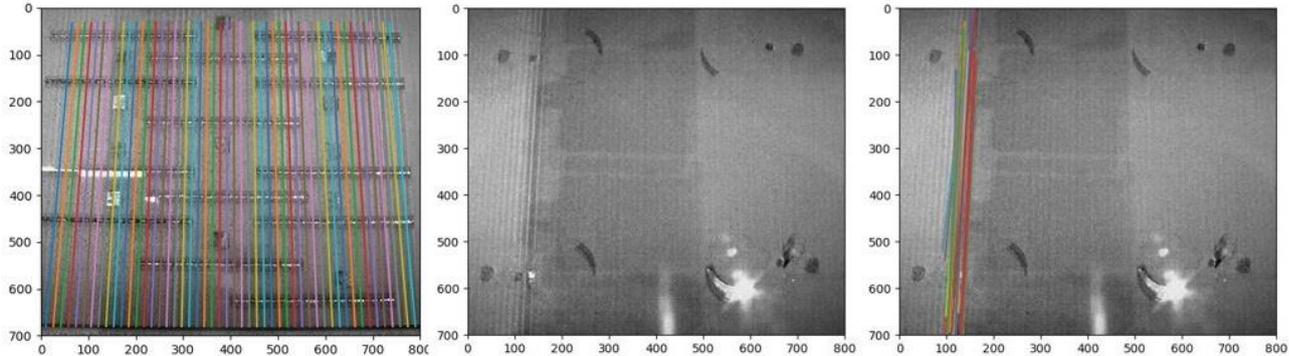


(fig. 7.17) Lines non horizontal or belonging to an object are excluded, leaving the defects.

*Waves*

The waves identification process is similar to the horizontal defect identification, as a progressive probabilistic Hough transform is used to identify the lines present in the segmented image, but the filtering of the result is slightly more complex due to the fact that the frame is deformed in the vertical direction (since the camera cannot be placed right above the middle of the building site, where the laser access the chamber): depending on the horizontal coordinate, a range of feasible angles is computed, and if the inclination of the identified line falls inside the range it is considered a wave defect.

To correctly identify the number of defects, a further step is required. Since the waves generated cover a wider area than the single horizontal defect (multiple parallel waves), the lines identified up until this point are grouped together through a Dbscan clustering, that contrary to the already used k-means clustering is based on distance, thus allowing to group together waves generated together (fig. 7.18).



(fig. 7.18) on the far left the reference angles depending on the coordinate, accounting for deformation. In the middle the target image, from which the lines are extracted (far right).

## Validation:

The complete algorithm is validated assembling a pool of images and manually extracting the defects feature to create a ground truth. The same pool is then fed to the algorithm to extract the features and compare the results (tab. 7.2), considering that some features required visual confrontation of the results, like in the case of the vector identification in the spattering function.

The pool of images is composed of 27 layers, where a layer consists of an image of the complete layer (with even number) and an image of the new layer (with odd number). Six images are skipped to test the capability of the algorithm to react to a missed acquisition. The images are taken from different manufacturing jobs, to increase the number of defects present, hence the significance of the image pool (tab. 7.1).

	holes	spattering	incandescence	horizontal	waves
Number of images showing defects	13	22	15	12	8

(tab. 7.1) Number of images showing a certain defect in the pool used for validation

layer	Holes						Waves								
	Ground Truth		Algorithm		Accuracy		Ground Truth			Algorithm			Accuracy		
	num	area	num	area	num	area	num	start	width	num	start	width	num	start	width
1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	1
2	2	102	2	107	1	0,95327	3	787	657	3	786	774	1	0,99875	0,85375
3							0	0	0	0	0	0	1	1	1
4	2	27	2	30	1	0,9	0	0	0	0	0	0	1	1	1
5	2	10	0	0	0	0	2	280	279	2	291	290	1	0,98625	0,98625
6	0	0	0	0	1	1	1	297	279	1	270	256	1	0,96625	0,97125
7	1	86	1	90	1	0,95556	0	0	0	0	0	0	1	1	1

Accuracy	FP	FN
0,79828	0,8021	2

Accuracy	FP	FN
0,89583	0,98369	0,98847

(tab. 7.2) comparison of ground truth and results of the algorithm. The lines marked in grey correspond to a missing image, artificially inserted to test the ability of the algorithm to detect missing acquisitions.

Since most of the algorithms relies on a correct contouring to correctly identify the defects, the function responsible for contouring is also validated.

A pool of 51 images is created containing objects, and then manually segmented to create the ground truth. The same pool is then fed to the active contouring algorithm, and the Sorensen-Dice coefficient is computed for each sample. The mean value expresses the accuracy of the designed algorithm.

The results of the validation will be discussed in the chapter “Conclusion and further developments”.

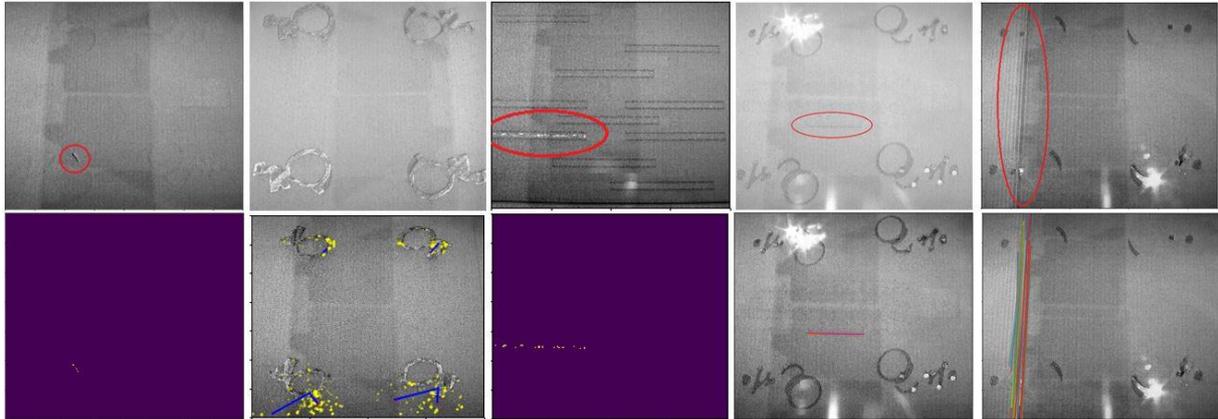
## 8. Conclusions and further developments

The conclusion of the thesis will be presented in three parts: at first it will be summarized what has been done for the acquisition of the images, then the results of the analysis will be discussed, separating the single algorithms for feature extraction working separately and the whole algorithm altogether. Finally, further development will be proposed, in a context of exploiting the data for a predictive maintenance process.

The acquisition system, as already described in the dedicated chapter, consist of managing the data flow coming from preexisting hardware. A sensor has been build exploiting a photoresistor to read signals from the machine electrical panel, while the access to the camera to perform the acquisitions and the management of said signal is perform via python scripts. The optimization of the hardware infrastructure, achieved by implementing an Arduino Uno board and a PC, allowed a higher precision, mainly because of the analog inputs of the board and the fact that being a microcontroller no other process interfere with the execution of the routine: the issues regarding the images are related to imprecision in the timing of the captures, either by capturing the recoater in the frame, or the laser in action. Once the optimization is in place, the number of bad timed acquisition in a whole job is 5/4658 (0.11%) in which four are the result of a slightly fast acquisition on the new layer, causing the shadow of the recoater to appear in the frame, possibly causing a false positive, while the other one is the result of a slightly slow acquisition, resulting in the laser to appear in the shot. This, however, doesn't cause any error until a significant portion of the piece is manufactured, issue that never occurs. The optimization solves via some issues that in previous versions ultimately led to error in the analysis.

The analysis algorithms are initially tested as standalone, tuning the parameters and visually checking the results. The results are positive, as all the defects are correctly identified, provided the requirements are met: the object needs to be identified with sufficient precision for the identification of the spattering, the horizontal defects and the incandescence in the artifacts, as these defects can only be present entirely outside or entirely inside the object. The waves defects are present in the entire length of the building site, while the holes in the powder bed appears before the layer is manufactured.

The evaluation of the single algorithms is done with a visual comparison of the original image and the resulting image on which the features are extracted, as it is the simplest way of quickly check the effectiveness of the algorithms (fig. 8.1).



(fig. 8.1) visual identification of a defects' features.

Once the single algorithms are tested and the complete algorithm is created by adding a normalization function to allow a precise tuning of the parameters, a validation is performed creating a pool of images that shows defects, listing these defects to create a ground truth and then executing the algorithm on the pool to compare the result and the a-priori knowledge on the data. Any false positive or false negative is noted, and then the accuracy is computed with the relative error of the measure with respect to the reference value (tab 8.1).

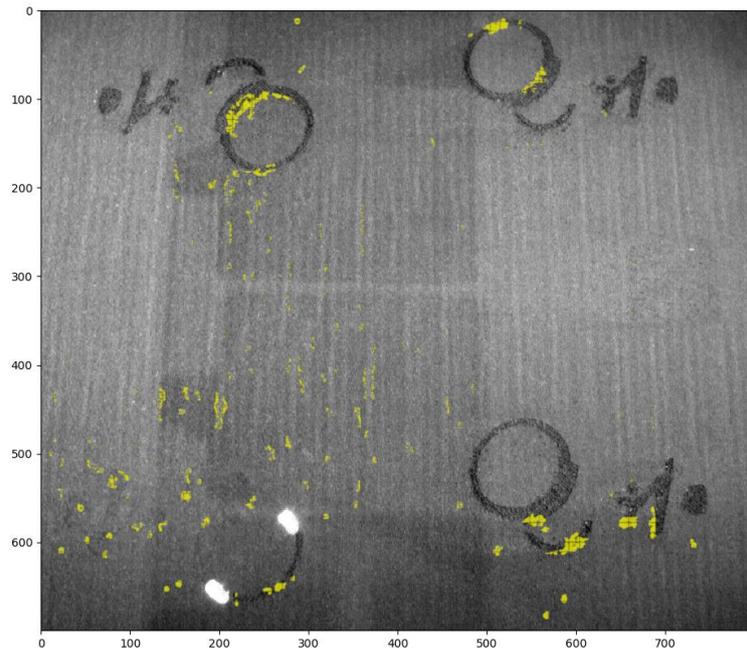
	Holes		Spattering		Incandescence		Horizontal	Waves		
	number	area	number	area	number	area	number	number	length	width
Accuracy	0,79828	0,80211	0,5641	0,4593	0,71129	0,59939	0,79954	0,89583	0,98369	0,98847
FP	2		1		5		1	0		
FN	0		0		0		1	2		

(tab. 8.1) accuracy results of the different algorithms on a pool with abundance of defects.

An important consideration needs to be done regarding the results of the validation. The image pool used is composed, to check for robustness, of some images that were the result of the acquisition process before the optimization of the hardware. The first effect of this choice is the presence of some false positive in the “holes” identification, and a measure that is much higher than the true value due to the presence of the recoater in the frame. The accuracy without considering that case rise to 96.49% for number of defects and 92.08% for area measurement. Another result of these inaccuracy is the difficulty in object contouring, as the images in which the laser is present have at least one area in which the intensity is much higher than the surrounding. The snake used for contouring is instead designed to be attracted by darker region, thus creating error in handling the laser area.

Both these issues are solved with the optimization in the acquisition process.

It’s worth discussing some issue with the spattering identification algorithm. The logic behind the process is the identification of darker regions with area inside a feasible range, that are not part of any object. An observed issue is the impossibility for the algorithm to distinguish the spattering from other defects with the same characteristics, like the horizontal defects or the waves (fig. 8.2). The accuracy of spattering identification in images with correct contouring and no other defects present is in fact 62.44%, higher than the computed accuracy.



(fig. 8.2) it is shown how the spattering identification follow the vertical lines resulting from a wave presence.

It is also noted how, due to the high number of distinct particles, the sensitivity to change in area value is much higher than features with reduced number. In this case a visual confrontation between an output image from the algorithm and the original capture might be more significant in assessing the efficiency of the algorithm itself. The contouring function is also subject to validation, and the medium value of the Sorensen-Dice coefficient is 0.7448, with values as high as 0.93 and a few as low as 0.5, in the images in which the acquisition had the laser inside the frame, as confirmed by visual confrontation of the ground truth and the contoured images. About the performances of the script, the algorithms for feature extraction taken individually don't take a lot of time to be executed, compared to the execution time of a layer, while the functions used to improve precision such as active contouring and normalization are more time consuming (tab. 8.2).

Operation	min execution time [s]	max execution time [s]
Holes identification	0,79	0,87
Spattering identification	0,57	0,72
Horizontal identification	0,59	0,86
Incandescence identification	0,82	0,95
Waves identification	0,93	4,67
Mask creation (normalization)	117,43	117,59
Normalization	15,57	16,81
Active contouring	44,41	496,25

(tab. 8.2) Execution times using a 4 cores computer running Ubuntu 18.04.3 LTS

The first improvement proposed is a richer communication between functions, accounting for issues derived by the contemporary presence of multiple defects, like the aforementioned interference between spattering and horizontal defects. An even more degree of precision could be achieved by accessing the source files for the manufacturing (CAD and slicing), allowing to compare the acquisitions with the theoretical shape of the layer, also creating a contouring directly from the design of the pieces.

During the thesis project, one of the monitored jobs was the subject of some mechanical tests for the characterization of a new material. A very interesting follow-up on the project is linking the change in mechanical properties with the appearance of defects during the process, as it was done for other properties, through machine learning algorithms. A control structure could also be implemented for predictive maintenance, once a correlation between process parameters and defect appearance is found, that can result in a significant decrease of the latter. To achieve such a program, an online version of the algorithm is most probably required: the immediate improvement from the current program is therefore the optimization of the algorithms to extract the features in a more performing time, while the references (like the background image use for normalization) would need to be provided at the beginning of the cycle or as external references. The most critical functions are currently the active contouring and the normalization, although for most of the manufacturing cycle (excluding the fastest final layers) the normalization can still be considered feasible, thus it is recommended the implementation of a segmentation for object identification.

The thesis project allowed the creation of a data collection system, a pivotal component of today's most innovative applications, and in this case the first step towards the development of a sophisticated control infrastructure for the most recent breakthrough in the manufacturing industry, the Additive Manufacturing.

## 9. References

1. Wang, D.; Wu, S.; Fu, F.; Mai, S.; Yang, Y.; Liu, Y.; Song, C.; Mechanism and characteristics of spatter generation in SLM processing and its effect on the properties [Mater. Des. 117, 121-130]. Amsterdam. Elsevier, 2017;
2. Heeling, T.; Gerstgrasser, M.; Wegener K.; Investigation of selective laser melting spatter characteristics for single- and multi-beam strategies using high speed imaging, LIM 2017;
3. Zhang, B.; Li, Y.; Bai, Q.; Defect formation mechanisms in selective laser melting: a review. Berlin. Springer-Verlag, 2017;
4. Gu, D.; Shen, Y.; Balling phenomena in direct laser sintering of stainless-steel powder: metallurgical mechanisms and control methods [Mater. Des. 30, 2903-2910]. Amsterdam. Elsevier, 2009;
5. Bi, J.; Lei, Z.; Chen, Y.; Chen, X.; Qin, X.; Tian, Z.; Effects of process parameters on formability and surface quality of selective laser melted Al-Zn-Sc-Zr alloy from single track to block specimen [Opt. and Laser Tech. 118, 132-139]. Amsterda. Elsevier, 2019
6. Kass, M.; Witkin, A.; Terzopoulos, D.; Snakes: active contour models. Boston. Kluwer Academic Publisher, 1987;
7. Galambos, C.; Matas, J.; Kittler, J.; Progressive probabilistic Hough transform for line detection. IEEE Computer Society conference on Computer Vision and Pattern Recognition (Cat. No PR00149), 1999;
8. Caltanissetta, F.; Grasso, M.; Petrò, S.; Colosimo, B.M.; Characterization of in-situ measurements based on layerwise imaging in laser powder bed fusion [Add. Manuf. 24, 183-199]. Amsterdam. Elsevier, 2018;
9. Alldredge, J.; Slotwinski, J.; Storck, S.; Kim, S.; Goldberg, A.; Montalbano, T.; In-situ monitoring and modeling of metal AM powder bed fusion. AIP Conference Proceedings 1949, 020007, 2018; [<https://doi.org/10.1063/1.5031504>];
10. Grasso, M.; Colosimo, B.M.; A statistical learning method for image-based monitoring of the plume signature in laser powder bed fusion [Rob. and Comp. Integr. Manuf. 57, 103-115]. Amsterdam. Elsevier, 2019;
11. Yang, L.; Hsu, K.; Baughman, B.; Godfrey, D.; Medina, F.; Menon, M.; Wiener, S.; Additive manufacturing of metals: the technology, materials, design and production. Gewerbestrasse, CH. Springer International Publishing AG, 2017;
12. Gibson, I.; Rosen, D.; Stucker, B.; Additive manufacturing technologies. New York, Springer Science+Business Media, 2015;
13. Gebhardt, A.; Hotter, J.; Additive manufacturing; Munich, Hanser Publisher, 2016;
14. Radi, B.; El Hami, A.; Forming processes; London, ISTE ltd., 2016;
15. "Sintering". *The Encyclopedia Britannica*. The Encyclopedia Britannica, 2019 [<https://www.britannica.com/technology/sintering>];
16. Renishaw plc; Scheda di sicurezza (SDS) Aluminium powder AlSi10Mg; Staffordshire, 2017; K-5800-3619-01-A\_IT;
17. Murr, L.E.; Gaytan, S.M; Ramirez, D.A.; Martinez, E.; Hernandez, J.; Amato, K.; Shindo, P.; Medina, F.R.; Wicker, R.B.; Metal fabrication by Additive Manufacturing using laser and electron beam melting techniques; Shenyang, Journal of materials science and technology, 2012;

18. Gong, H.; Rafi, K.; Gu, H.; Ram, G.D.J.; Starr, T.; Stucker, B.; Influence of Defects on Mechanical Properties of Ti-6Al-4V Components Produced by Selective Laser Melting and Electron Beam Melting; Amsterdam, Elsevier Ltd., 2015; [<https://doi.org/10.1016/j.matdes.2015.07.147>];
19. EOS GmbH - Electro Optical System; 2016; M290 User manual; Munchen, EOS GmbH - Electro Optical System;
20. Wallace, J.; Wang, M.O.; Thompson, P.; Busso, M.; Belle, V.; Mammoser, N.; Kim, K.; Fisher, J.P.; Siblani, A.; Xu, Y.; Welter, J.F.; Lennon, D.P.; Sun, J.; Caplan, A.I.; Dean, D.; Validating continuous digital light processing (cDPL) additive manufacturing accuracy and tissue engineering utility of a dye-initiator package; Bristol, UK, IOP Publishing Ltd., 2014;
21. Masood, S.H.; Mau, K.; Song, W.Q.; Tensile properties of processed FDM polycarbonate material; Materials Science Forum (Volumes 654-656), 2010;
22. Zhang, K.; Fu, G.; Zhang, P.; Ma, Z.; Mao, Z.; Zhang, D.Z.; Study on the geometric design of supports for overhanging structures fabricated by Selective Laser Melting; Basel, CH, MDPI, 2018;
23. CustomPartNet; Olney, MD; 2008, [<https://www.custompartnet.com/wu/fused-deposition-modeling>];
24. Sidambe, A.; Biocompatibility of advanced manufactured titanium implants - a review; Basel, CH, MDPI, 2014;
25. EOS GmbH - Electro Optical System; 2016; M290 datasheet; Munchen, EOS GmbH - Electro Optical System [<https://www.eos.info/eos-m290>];
26. IDS Imaging Developing Systems GmbH; 2019; UI-1540SE-M-GL datasheet; Obersulm, IDS Imaging Developing System GmbH.
27. Canny, J.; A computational approach to edge detection [IEEE transaction on pattern analysis and machine intelligence, vol. PAMI-8 no. 6]. 1986;
28. [https://scikit-image.org/docs/dev/auto\\_examples/segmentation/plot\\_watershed.html](https://scikit-image.org/docs/dev/auto_examples/segmentation/plot_watershed.html)
29. <https://docs.scipy.org/doc/scipy-0.16.0/reference/generated/scipy.ndimage.measurements.label.html>
30. [https://scikit-image.org/docs/dev/api/skimage.segmentation.html#skimage.segmentation.active\\_contour](https://scikit-image.org/docs/dev/api/skimage.segmentation.html#skimage.segmentation.active_contour).