Creating a plenoptic camera through an external mask

Valls Farré, Lara
Curs 2015-2016

Director: JUAN FCO. GARAMENDI

GRAU EN ENGINYERIA EN SISTMES AUDIOVISUALS

Treball de Fi de Grau
Creating a plenoptic camera through an external mask

Lara Valls Farré

TREBALL FI DE GRAU
ENGINYERIA EN SISTEMES AUDIOVISUALS
ESCOLA SUPERIOR POLITÈCNICA UPF
2016

DIRECTOR DEL TREBALL
Juan Fco. Garamendi
I want to express my gratitude to my family; they have allowed me the chance of studying. Thank you for teaching me the value of the effort: “You cannot stop the waves but you can learn how to surf”.
Acknowledgements

I would like to thank Juan Fco. Garamendi for all his dedication. He has been such a great thesis supervisor. I want to be grateful for the amount of hours he has dedicated to me, for all his tips and implication. When I have been stuck I could count on you, to guide me. You have been a crucial support for the elaboration of the work. Thank you.

I also owe a huge thank to my family for their unconditional support and patience. Thank you for all the times you have believed in me. Also I would like to be grateful to my partner Javi for listening and encouraging me. Finally I would like to end by mentioning my friend Héctor, we have remained joined throughout these 4 years. Moltes gràcies a tots!
Abstract

Nowadays, there are plenoptical cameras through which can be obtained images that give us special information.
On each image, we get the direction from which the light reflected by the objects comes (lightfield). This type of information is useful in applications of computer vision such as estimation of the depth.
The purpose of this work, is the theoretical study and the creation of a plenoptical camera, departing from a conventional reflex camera without modifying any internal parameter of the camera.
The theoretical aspects that have been addressed have been the onset of vignetting, change of perspective and the design of the external mask. The working paper is complemented by the implementation of a method of depth calculation through the images we get with the camera created.

Keywords: plenoptical camera, light filed images and vignetting.

Resumen

Hoy en día, existen cameras plenopticas mediante las cuales se pueden obtener imágenes con información de la dirección de la que proviene la luz reflejada por los objetos (lightfield). Este tipo de información es útil en aplicaciones de visión por computador como la estimación de la profundidad.
El propósito de este trabajo es el estudio teórico y la creación de una cámara de este tipo partiendo de una cámara reflex convencional sin modificar ningún parámetro interior de la cámara. Los aspectos teóricos abordados han sido la aparición del viñeteado, cambio de perspectiva y el diseño de máscaras externas. El trabajo se complementa con la implementación de un método de cálculo de profundidad a partir de imágenes obtenidas con la cámara creada.

Palabras clave: cámara plenoptica, lightfield y viñeteado.
# INDEX

<table>
<thead>
<tr>
<th>Abstract</th>
<th>ix</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Chapter 1</strong></td>
<td></td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Context</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Motivation</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Objectives</td>
<td>3</td>
</tr>
<tr>
<td>1.4 Report structure</td>
<td>3</td>
</tr>
<tr>
<td><strong>Chapter 2</strong></td>
<td></td>
</tr>
<tr>
<td>2 STATE OF THE ART</td>
<td>5</td>
</tr>
<tr>
<td>2.1 What is lightfield</td>
<td>5</td>
</tr>
<tr>
<td>2.2 Ways and devices to acquire lightfield</td>
<td>8</td>
</tr>
<tr>
<td>2.3 Lightfield applications</td>
<td>10</td>
</tr>
<tr>
<td><strong>Chapter 3</strong></td>
<td></td>
</tr>
<tr>
<td>3 DEVELOPMENT</td>
<td>11</td>
</tr>
<tr>
<td>3.1 Building a lightfield camera</td>
<td>11</td>
</tr>
<tr>
<td>3.1.1 What is vignetting?</td>
<td>11</td>
</tr>
<tr>
<td>3.1.1.1 Mathematical explanation of vignetting</td>
<td>16</td>
</tr>
<tr>
<td>3.1.2 Mathematical demonstration between the engagement vignetting and</td>
<td>19</td>
</tr>
<tr>
<td>perspective</td>
<td></td>
</tr>
<tr>
<td>3.1.2.1 Vignetting</td>
<td>19</td>
</tr>
<tr>
<td>3.1.2.2 Perspective</td>
<td>24</td>
</tr>
<tr>
<td>3.1.3 Geometrical distance between limit light rays</td>
<td>30</td>
</tr>
<tr>
<td>3.2 Building a lightfield image</td>
<td>33</td>
</tr>
<tr>
<td>3.2.1 Design of the external mask</td>
<td>33</td>
</tr>
<tr>
<td>3.2.2 Shape, size and position of the holes</td>
<td>36</td>
</tr>
<tr>
<td>3.2.2.1 Shape</td>
<td>36</td>
</tr>
<tr>
<td>3.2.2.2 Size</td>
<td>36</td>
</tr>
<tr>
<td>3.2.2.3 Position</td>
<td>39</td>
</tr>
<tr>
<td>3.2.3 Camera preparation</td>
<td>41</td>
</tr>
<tr>
<td>3.3 Estimating depth</td>
<td>43</td>
</tr>
<tr>
<td>3.3.1 Data initialization</td>
<td>45</td>
</tr>
<tr>
<td>3.3.2 Optical flow calculation</td>
<td>48</td>
</tr>
<tr>
<td>3.3.3 Data processing</td>
<td>49</td>
</tr>
<tr>
<td>3.3.4 Results and limitations</td>
<td>51</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4 CONCLUSIONS</td>
<td>63</td>
</tr>
<tr>
<td>References</td>
<td>65</td>
</tr>
<tr>
<td>Annex</td>
<td>67</td>
</tr>
<tr>
<td>I. Pseudocode</td>
<td>67</td>
</tr>
<tr>
<td>II. Lightfield images</td>
<td>70</td>
</tr>
</tbody>
</table>
Chapter 1

INTRODUCTION

1.1 Context

Joseph-Nicéphore Niépce [1] will be proud of his research if he could see that nowadays in the XXI century, thanks to his idea of freezing a moment in time on a paper, we can enjoy of having technology such as plenoptic cameras. Niépce was a French inventor who captured the first permanent photograph of the nature on 1826.

If we move forward in time we find the origin of the beginning of the concept of plenoptic cameras on 1908, when the also French Gabriel Lippman came up with a new idea. His idea was to use an array of tiny lenses to project a scene onto a single sheet of film. The multiple views these lenses recorded could then be reconstituted into a 3-D image by viewing the processed film through an identical lens array. Three years later, Russian physicist P.P. Sokolov constructed the first integral camera using a pinhole array instead of the harder-to-fabricate lenses that Lippmann envisioned. However, building a plenoptical camera, required technologies that would not be realized for almost another century [2].

Now, the reader probably is wondering what is the difference between a conventional camera and a plenoptic one. In a conventional camera, the ray lights reflected from the objects of a scene are concentrated by the lens into a (electronic or chemical) sensor. This forms the photograph, keeping only information about intensity information. Any directional or angular information from light is lost. As a result, the image of the scene we get is a photograph of two-dimensions. We lost the depth information, and trying to estimate distance of the objects with respect to the camera is a very hard computer vision task, being necessary the use of perceptual strategies, such as dimensions of the object, occlusions, convexity, symmetry, parallel lines, etc. [3]. Besides, for the physical structure of the conventional cameras, the image we have recorded will have a fixed point of focus that has been determined by the lens and the diaphragm aperture when the photo has been shot.

Lightfield photography is more ambitious. This type of cameras not only resign it selves in recording all the intensity of every incoming light ray; moreover, it records the directional information of each ray. Thanks to having the directional information of each light ray, after the photograph has been recorded, the information can be manipulated combining the light information such a way can be done at the acquisition moment. As result, some parameters of the acquisition as aperture or position of the camera can be – slight- modified in a post-processing step, i.e. We can get a collection of images taken from different point of views or we can change the depth of field or the focus points.

But as Winston Hendrickson, vice president of engineering for digital imaging at Adobe Systems says: “Plenoptics is about way more than refocusing images. In capturing all the spatial and angular information about a scene, you can do things like motion parallax, changing the perspective, and detecting objects” [2].
Smartphones have transitioned into powerful computing devices. Users will realize the ability to capture 3D images and video from their mobile device, giving them unprecedented freedom to produce their own 3D content for a wide range of applications: 3D selfies, virtual and augmented reality, 3D printing, gaming, etc. According to IDC, by 2018, 85% of the global image capture volume will come from mobile devices. Users increasingly want more features and functionality from the camera in their mobile phone, because that’s the ever-present camera in their pocket. The team at Pelican Imaging has developed revolutionary new array sensor technology that captures in 3D. Pelican's depth-sensing array solution provides highly accurate depth data in real time, both for images and video [4]. And all of this it is thanks to lightfield images; this is just one type of plenoptic cameras that now a day exist. The team pelican Imaging have incorporate an array of micro cameras on the phones in order to take benefit of what lightfield brings us. Moreover, we can achieve lightfield through a single camera with the help of a physical structure, or with an array of cameras and furthermore with the introduction of an array of micro lenses in front of the sensor of a conventional camera. We will explain that in more detail in the next section. However all of them present a problem: the prize. As it is a recent technology that is just at the beginning of its growth, now a days it does not exist already a low cost camera in order to achieve lightfield. In this project, we have proposed the challenge to create a plenoptic low cost camera.

This project is based on the article “External mask based depth and light field camera” done it by Dikpal Reddy, Jjamin Bai and Ravi Ramamoorthi. On this article, they present a method to convert a digital single-lens reflex (DSLR) camera into a high resolution consumer depth and light field camera by affixing an external aperture mask to the main lens [5]. However, in the article the basic mathematical proofs of the parameters that we have to take care of, is not presented in extensor. On this thesis, we have done a rigorous guideline where we have studied and demonstrate mathematically step-by-step all the process in order to achieve a light field camera understanding all the mathematical concepts.

1.2 Motivation

During all this four years in the career of engineering on audiovisual system I have been really interested in a particular topic, image processing. Of that interest is awakened the ambition to develop a project on image processing that allowed me the chance to use all the knowledge that I have achieved during this four years. I have the desire to develop a project that demonstrates myself that all the effort during the career has been worth it. Then, and thanks to the subject video processing, I have discover all the wide range of things we could do with lightfield images.
1.3 Objectives

Apart from the personal satisfaction, the main objectives treated in this Msc. thesis, are the following ones:

- To convert a conventional reflex camera in a plenoptic camera without changing any internal parameter of the camera.

- To mathematically describe the parameters of our plenoptic camera.

- Solve some of the computer vision problems in literature with our plenoptic camera.

1.4 Report structure

This Msc. thesis is organized in four different parts. The first one provides a quick overview about the thesis in order to get the reader in context and with the desire to captivate them with the topic. **Chapter 2** introduces some basics concepts in order to be able to understand the entire thesis. This chapter has the aim to give the reader all the key concepts that will be helpful for the properly understanding of the following chapters. **Chapter 3** presents all the implementation and study that have been carried out in order to achieve all the objectives. This chapter contains all the mathematical demonstrations that have been required to study. Moreover, contains the guideline to be able to construct the properly design of the external mask. Finally we can see an example of the complete implementation of achieving lightfield images through a conventional reflex camera with the help of an external mask. Finally, in **Chapter 4** we can see the conclusions we have obtained after doing this entire project, as well as the future work that should to be done.
Chapter 2

STATE OF THE ART

2.1 What is light field?

Our humanity lives in a world which it is illuminated from different light sources, the main ones: the sun. This central star of the solar system radiates light for overall the galaxy making our world capable to shine. The directional light rays we get from the sun hit on the earth’s surface reflecting the rays in every direction. And we can perceive our reality thanks to this. We can contemplate a landscape or interact with an object thanks to this directional light rays that collides with them creating one continuous light field.

For our biological structure, we do not see all the infinite light rays form this light field. We just can see all the light rays that pass through our pupils and concentrated in one point of the retina. Moreover, when we look to an object, we are able to see this object from two different perspectives without moving our head. We get the first one closing the right eye and we get the second one closing the left eye and viceversa. Thanks to the different spatial location of our eyes, we can have different perspectives and as a result our brain is able to realize depth estimation.

A camera is a device used to capture images or photographs, like our eyes do but with a slightly difference. Cameras can only capture the scene, from one perspective whereas our eyes, we can register two direction, one of each eye.

The light rays of the scene, passes through the lens until arrive to the sensor in order to record the image, getting the information of color light and intensity. Cameras could not get any angular information.

In the following images, we can see an apple as the scene that we want to capture with a conventional camera. We also can appreciate that the light rays coming from a light source is hitting the apple and thanks to that, as we have said, we are able to visualize it.

The photography of this apple will contain the color and intensity information, but none any directional information, as we only are getting just one point of view.

Figure 1 Representation of a camera recording a scene of an apple. The apple is hit for many light rays of a light source. As the camera it is a conventional one; it only captures the light ray intensity and color. Source¹: LYTRO.

¹ http://blog.lytro.com/post/132599659620/what-is-light-field
If we go a little further, we found stereo cameras. Having the same scene than on the images above, we can see one clear difference. With this camera, we are recording the color and the intensity and besides one more point of perspective rather than before. With stereo cameras, we are getting two different perspectives.

![Figure 2](http://blog.lytro.com/post/132599659620/what-is-light-field)

**Figure 2** Representation of a camera recording a scene of an apple. The apple is hit for many light rays of a light source. As it is a stereo camera, it captures the light ray intensity, color and two light directions. Source²: LYTRO

In contrast to conventional cameras, a light field cameras, also known as plenoptic cameras, allows light to be captured from multiple directions. There are many ways to capture light from multiple direction, we will focus on that on the next section 2.2 Ways and devices to acquire lightfield. But on the following image, we can see one way of getting multiple light directions. We can appreciate that unlike conventional cameras, an array of micro lenses is located in front of the sensor.

![Figure 3](http://blog.lytro.com/post/132599659620/what-is-light-field)

**Figure 3** Representation of a camera recording a scene of an apple. The apple is hit for many light rays of a light source. The camera is plenoptical, as it has an array of micro lenses in front of the sensor. With this type of camera, we can capture the light ray intensity, color and angular information. Source³: LYTRO

---

Staying the camera in a fixed location, for each lens of the array we are getting a different perspective of the same scene. We can see that on this image, where with one sub-image the apple is occluding the right part of the orange and on another sub-image is occluding the left part.

![Diagram of lightfield camera](image)

**Figure 4** We can see the array of micro lenses that split the light rays after they arrive to the sensor. This way, as we can see under, we generate different images from the same scene that have slightly different points of view. Source⁴: LYTRO

Having recorded different light ray directions, will give us much perspectives us many directions we get. As a result, of having different perspective, light field cameras, furthermore to capture color and intensity information we will get angular direction information that will allow us to produce depth information.

The essence of Lightfield relies in the concept that the light that arrives from any point of the scene, owns all the information we need in order to reproduce any view of this scene with this camera location. In other words, when we want to take a photography and we desire to emphasize on element of the scene making the background blurred for example, we have to adjust the lens of the camera in order to achieve this effect. Or maybe we prefer to emphasize the background and we want that the elements that are closer to the camera appear blurred and we have to readjust the lens. With lightfield this problem no longer exists, because we have all the light information that is necessary to reproduce these two different views.

The information a light-field camera records is, mathematically speaking, part of something that optics specialists call the plenoptic function. This function describes the totality of light rays filling a given region of space at any one moment. It’s a function of five dimensions, because you need three \( x, y \) and \( z \) to specify the position of each vantage point, plus two more the angle of every incoming ray [2].

2.2 Ways and devices to acquire lightfield

As we have said in the section before, 2.1 What is lightfield, in order to generate lightfield images, we need to have slightly different points of view from the same scene. Nowadays, in order to acquire lightfield images, we have basically four ways.

The three first ways, does not need any external or internal change of the conventional cameras.

On one hand, we could achieve lightfield images with a single conventional camera. With the assumption that the scene we want to record is static, we could generate lightfield images by moving the single conventional camera through the entire scene. We will have to estimate and study the locations of the camera have to be in order to achieve the properly different points of view. We will need a kind of physic structure in order to move the camera in an accurate and precise way.

On the other hand, we can also generate lightfield images with an array of conventional cameras. Every camera of the array must be carefully synchronized. In other words, it has to take care of how they are orientated, how we adjust the focal length, the shutter timing and the exposure. All of this is really important because we want to generate an array of images of different points of views but all these images have to share the same light and the same point of focus. With this method and unlikely to the method of the single camera, the scene we want to record, could be dynamic.

Nowadays, it already exist a Smartphone of the company Pelican that incorporates an array of micro cameras. As we can see on the right on Image 5 we can appreciate many different lens that behind it we found different sensors resulting as an array of micro cameras. Pelican's depth-sensing array solution provides multiple benefits for AR applications, including highly accurate near-field and far-field depth data captured in all lighting environments, real time processing, dynamic calibration, and minimal occlusion zones. The layout of the array is flexible and can be customized to meet specific depth resolution requirements [4].

Figure 5 We can see three images that will be explained from left to the right. The fist one is an example of the physical structure that we will need to make the accurate movements, in order to create lightfield images with a single camera. The second one, we can appreciate an array of cameras that as they are located in a different place, each camera will record a slightly different perspective of the scene. Finally, on the right, we find one Smartphone that its camera consist in having many lenses with different sensors converting it in an array of micro cameras

Source[5], 6: Article of Light Fields and Computational Imaging and Cnet

Moreover, the other two methods could not be done with a simple conventional camera. On the first case we have to introduce an array of micro lenses inside the camera and on the other we have to put an external mask outside the camera. Let’s focus our attention on the first case. This is perhaps, the most popular and efficient way to create lightfield images. However each camera prize is really expensive. Nowadays according to the company *quesabesde* [6] the cost of plenoptic of one of the pioneer companies in this sector, Lytro, could have accost of 1.600 Euros.

Lightfield cameras, or plenoptical cameras, inside the camera have an array of micro lenses located just in front of the sensor. This array as the name suggests, consist of many microscopic lenses with tiny focal lengths. What do this array of micro lens is separate light ray directions before the light arrives to the sensor. And each image will differ a little bit from the rest, because every lens is getting different light rays directions from the neighbor lens

In this type of plenoptic cameras, we find two subdivisions. This subdivision depends on the exact location of the array of micro lenses.

In the first subdivision called plenoptic 1.0, is illustrated on *Figure 6*, we find the array of micro lens in front of the sensor, as we have already said and furthermore, in the focal plane. The main lens focuses all rays in the focal plane, where we have put our array of micro lens. What the array of micro lens does is to split the rays. Hence, for each pixel before the micro lens, we have a different point of view.

Secondly, the other subdivision, called pleoptic 2.0, is illustrated on *Figure 7*, we find the array of micro lens before the focal plane. The main lens concentrates all the rays in the focal plane and to continue, the rays begin to disperse. In this moment, we find the array of micro lenses that where it gets all the light beams. Then each micro lens joins again the entire ray light in a unique point. Hence each point will have a different point of view [7].

---

**Figure 6** Simple representation of the internal functionality of a plenoptic camera that has the array of micro lens on the focal plane

**Figure 7** Simple representation of the internal functionality of a plenoptic camera that has the array of micro lens before the focal plane
Finally the last way to achieve lightfield images is by using an external mask. It will consist on making an external mask where this mask will have different holes. For every hole, we will capture an image and we will change the hole position before taking the next image. It differs from the previous method for not having to introduce or change anything inside the camera; hence we can convert one conventional camera into a plenoptic camera. The function of the mask is to block some light ray directions. As a result, every image, will have a slightly change of perspective because they will have different light ray directions. In this case, again, we need to have the assumption of that the scene is static. This process is the cheapest one because we can do so with any conventional camera and we do not need more than one camera; but also it will have some limitations, such as we cannot capture in one shoot all the images because we need to generate as much photographs as much holes our external mask have.

### 2.3 Lightfield applications

The fact of having in our pictures angular information is useful in applications of computer vision. Through a software, we can process all the lightfield images in order to relate them and having some benefits that in a conventional camera could not be changed after we have recorded the image. Using any of the four methods mentioned on the section above, 2.2 Ways and devices to acquire lightfield, in order to obtain lightfield images we can define which part of the image we desire to be in focus or out of focus, we also can change the depth of field that our images have. Moreover, we can shift the perspective. What is relevant is that we can do all this things after the photography has been shooted because the images contain depth information.

Maybe the reader is wondering for example, how we can change the focus point after the photography has been shot. Let’s remember that with a conventional camera we have to adjust the lens in order to make focus where we desire; so that all the light rays coming from one point on your subject converge at one point on the camera’s sensor. Depending on whether the subject is near or far, you move the lens in or out to achieve proper focus. A lightfield camera, if we are on the case that we have an array of micro lens in front of the sensor, doesn’t need to move its lens, because it can calculate the lightfield at any plane inside the camera. So it can generate images corresponding to various separations of lens and sensor, from close-ups to views of the distant horizon. But capturing the full light field has a price. First, there’s a vast increase in the amount of data the camera must acquire. Second, there is a significant loss in resolution of the final image, which is effectively limited to the number of micro lenses rather than the resolution of the camera’s sensor [2].
Chapter 3

DEVELOPMENT

3.1 Building a lightfield camera

In this section, is explained all the parameters we have to take care of, if we desire to achieve lightfield images. Besides, it have been studied the reasons why this parameters appear and it have been demonstrated it empirically and mathematically.

3.1.1 What is vignetting?

The effect of vignetting is that on the image we get, most part of the scene is not appearing. In other words, we obtain an image where we only can see one little part, because the rest is black. As we can see on Figure 12 and 13. All of this is occurring because of putting an external mask.

Considering a 1D scenario, Figure 8, we can parameterize the image as a 2D function $L(u,s)$, where ‘u’ is the coordinate in the sensor plane and ‘s’ is the coordinate on the lens plane, as we can see on Figure 14.

![Figure 8](image)

*Figure 8* When we put an external mask in front of the objective of the camera with a distance ‘x’, the effect that is generating this fact, mathematically, it is equivalent of having a virtual mask behind the sensor with a distance ‘y’. For simplification, we will work with this virtual mask for understanding what vignetting is.

On figure 8.1, we have represented a 1D scenario where we have a green point on the focus plane. Moreover, we have highlight six rays. Each ray is passing in a different position of the lens, the red one is passing through ‘s1’, the green ray through ‘s2’, and
so on. As the point is on focus, all rays are converging on the same location on the sensor: ‘u1’.

Figure 8.1 Representation of a 1D scenario where we have an object on the focal plane and as a consequence all its rays converge in on point on the sensor

Now, if we translate this 1D scenario in a Cartesian plane with coordinates ‘u’ and ‘s’, we obtain Figure 8.2. As it is expected, for each coordinate of ‘s’, will have the value of ‘u1’. Then as a result, we will draw a schuss.

Figure 8.2 Translation of the Figure 8.1 on a Cartesian plane
Now, we have the same case with a slightly difference. The object it is not on focus because it is not located on the focal plane. As a result, all light rays will converge however they will do it before the sensor. Hence the object will be blur. If we pay attention, we will see how the red ray that passes through the position of ‘s1’ in the sensor, is arriving in the position of u6 in the sensor. The green one is passing through ‘s2’ and arriving on ‘u5’ and so on.

Figure 8.3 Representation of a 1D scenario where we have an object is not on the focal plane and as a consequence all its rays converge in on point before the sensor.

If we make the same translation on the Cartesian plane, we obtain an inclined schuss. This is because for every value of ‘s’ we have a different value on ‘u’ axis.

Figure 8.4 Translation of the Figure 8.3 on a Cartesian plane
Once we have understood, what is representing our Cartesian planes, we can take a look on Figure 9, 10 and 11 in order to explain the effect of vignetting more technically.

These images are representing one simple scene with two points. The blue point is in focus and the green one is out of focus, so they are in different spatial locations. According to the axis, ‘u’ is the position of the sensor and ‘s’ is the coordinate where passes the light ray. Moreover, ‘m’ it is the position of the virtual mask.

On Figure 9 we can see that our sensor is receiving information for overall the points. It can be seen as the blue point, is on focus because for each coordinate of ‘s’ we obtain a constant value of the coordinate of ‘u’. However, for the green point this is not happening. And this is because this point is not sharp, then the value is not a constant. But what is relevant here is that our sensor is getting all the light rays of each point. This is denoting on the Figure 9 where all the line has the same intensity of color. This is the result of not having applied to our camera and external or internal mask and as a consequence not being blocking any light ray. On Figure 10 and 11, we will see that this is not happening because there, we have an external mask.

On Figure 10 that the external mask or the virtual mask is located on the optical center of the lens, hence we do not have vignetting. Let’s explain how we could know why the external mask, is located on the optical center. First of all we will pay attention on the blue point of the scene. As I have said, this point is in focus, so for each coordinate of ‘s’ we obtain a constant value of the coordinate of ‘u’- because this point is sharp-. Contrary, for the green point, for each coordinate of ‘s’ we obtain a slightly different value of the coordinate of ‘u’ because it is not in focus, and as a result is blur. Also we can see that in both cases, only one part of this line is darker. This is representing the fact that we have a mask that is only allowing to passing some rays. Where the color is more intense, is where our mask is not blocked. We can know that our external mask is located on the optical center because the fact of having a mask is not preventing that our sensor gets all the information. We will see more clearly on Figure 11.

If we take a look at Figure 11, we can see that now, the virtual mask is not on the optical center, so it will generate vignetting. Let’s understand why. The effect of vignetting, is the result of that our sensor has no information. In this case, for the position where the virtual mask is located, we can see clearly that in the orange shaded areas, the sensor has no information. As we have said before the only information that
we get is delimited with the red lines, because it is the light rays that the mask is allowing to enter. As a result the inclination of this schuss, will determinate the vignetting that we have. In other words, on Figure 8, as the schuss was horizontal, we have no vignetting because for each point of the sensor, we have information. But now on Figure 11, tiling this schuss, we are leaving areas with no information – the orange once-. As a result of having no information on the sensor, we have black areas.

![Figure 10](image1.png) **Figure 10** Representation of a scene with two points where only the blue once is in focus. Here we have the external mask on the optical center of the lens. As a result we do not obtain vignetting on our images

![Figure 11](image2.png) **Figure 11** Representation of a scene with two points where only the blue once is in focus. Here we have not the external mask located on the optical center of the lens. As a result we obtain vignetting on our images

Here we can see a pair of photographs, showing what will obtain on the case we take an image with the camera having an external mask which is not located on the optical center, and another one where we do not have any mask. We could not show the result of taking a picture where the mask is located on the optical center, because we will need to buy a plenoptic camera where the fabricant has put the mask inside.

![Figure 12](image3.png) **Figure 12** Photographs taken without any external or internal mask. It is related with Figure 7. We can see that we receive all the light rays for each object and moreover, our sensor is getting all the information thus we do not have any black area.

![Figure 13](image4.png) **Figure 13** Photographs taken with an external mask. It is related with Figure 9. We can see that we do not have all the light rays for each object. Furthermore, as a consequence of not have put the external mask on the optical center, our sensor have areas where we are losing information. As a result we have black areas. Moreover, on the areas that our sensor gets information, as a result of the external mask, we are also losing intensity
3.1.1.1 Mathematical explanation of vignetting

If we take a look at Figure 14, we can see the representation of the light ray crossing the virtual mask in order to arrive in the sensor. Specifically, this ray is the light ray that passes through the middle point of the aperture mask – ‘m’-.

Also we can see all the corresponding values for the process of shutting with an external mask.

If we focus our attention on the blue and orange triangle, represented on Figure 15 in order to only have the relevant information, we realize that we have a relationship of triangles. Then for similarity of triangles [8], we can arrive to the equation of:

\[
\frac{s-u}{m-u} = \frac{d}{d-d_m}
\]
If now, we operate in order to isolate our unknowns, we obtain the following result:

\[
\frac{s - u}{m - u} = \frac{d}{d - d_m}
\]

\[
s = -\frac{d_m}{d - d_m}u + m\left(\frac{d}{d - d_m}\right)
\]

\[
(s - u)(d - d_m) = (m - u)d
\]

\[
s = -\frac{d_m}{d - d_m}u + m\left(\frac{d + d_m - d_m}{d - d_m}\right)
\]

\[
sd - sd_m - ud + ud_m = md - ud
\]

\[
s = -\frac{d_m}{d - d_m}u + m\left(\frac{d - d_m}{d - d_m} + \frac{d_m}{d - d_m}\right)
\]

\[
sd - sd_m = -ud_m + md
\]

\[
s = -\frac{d_m}{d - d_m}u + m\left(\frac{md}{d - d_m}\right)
\]

\[
s (d - d_m) = -ud_m + md
\]

\[
s = -\frac{d_m}{d - d_m}u + \frac{md}{d - d_m}
\]

As we have seen, vignetting depends on the inclination of the red schuss represented on Figure 10 and 11. We have already seen that when this schuss is horizontal, we have information of overall the sensor. However, when this schuss has slope, we lose information in some areas of the sensor. Hence, as a result, we can see that vignetting is related to the inclination of the schuss.

\[
s = -\frac{d_m}{d - d_m}u + m\left(1 + \frac{d_m}{d - d_m}\right)
\]

We can rewrite it, where \(\rho = \frac{d_m}{d - d_m}\). This way, we will have the same equation more similar to the typical equation of the line \(y = mx + b\). As a result, it will be easier to identify the slope:

\[
s = -\rho u + m\left(1 + \rho\right)
\]

It is clear to see that the slope of this line is \(\rho\). So, depending on the value of \(\rho\) we will generate vignetting of not. As \(\rho = \frac{d_m}{d - d_m}\), where \(d_m\) is the distance between lens and the virtual mask and \(d\) is the distance between the lens and the sensor.
As a result, higher the value of $d_m$ is and as a consequence we are putting the external mask further away of the optical center, we will generate more vignetting because the value of $\rho$ will be higher.

On the other hand, if we are putting the external mask right in front of the optical center, so the value of $d_m = 0$, the value of $\rho = 0$ so the schuss will not have slope so we will have information for overall the sensor.

Now we have demonstrated mathematically the different results we get depending where we locate our external mask. Furthermore, we have checked it empirically.

In the following images, we can compare the vignetting that we get depending on the external mask we are using. These photographs have been taken on the same conditions – shape and size of the hole, zoom of the camera, focus point, etc.-.

This makes sense to us because the design of external mask used to take Figure 17, the cover was five centimeters far from the objective, so the value of $d_m$ was bigger than the other design. The design used for take Figure 16 was located so close to the optical center it was possible: right in front of the objective. This way the value of $d_m$ is more close to 0. So as a result the slope of the first case, it is bigger. For that reason we get more vignetting. Whereas in the second case our slope tends to be more horizontal despite it is not because we also have vignetting.
3.1.2 Mathematical demonstration between the engagement vignetting and perspective

3.1.2.1 Vignetting

First of all, we will focus in demonstrate mathematically the relation vignetting has with the aperture of the external mask hole. Lately, we will also explain this same relation according to perspective.

On Figure 18, we can see a mathematical modeling of what is happening in the camera. The value of $\alpha$ will affect also on the vignetting we get. The value of $\alpha$, is the geometrical distance between limit light rays that the external mask leave to pass. On the section 3.1.3- Geometrical distance between limit light rays, we will demonstrate that $\alpha = \Delta \sqrt{\frac{(1+\rho)^2}{1+\rho^2}}$.

On Figure 19 we can observe the same scene; however, the aperture of the external mask is much bigger. As we have already said, $\alpha = \Delta \sqrt{\frac{(1+\rho)^2}{1+\rho^2}}$, it is a constant, then if we make bigger the aperture of the external mask, we are increasing the value of $\Delta$. Hence, the value of $\alpha$ is much bigger too.

Intuitively, as we can see on Figure 18 and 19, the bigger $\alpha$, the bigger the field of view. So more are without vignetting is getting our sensor – we are getting more range of the horizontal coordinate $-$. As a result, we have less black zones on the image. The part that it is colored with orange represents the part of the sensor where it has no information, so we have vignetting.

More deeply explanation of this fact is found on the in the previous point: 3.1.1 What is vignetting?

![Figure 18](image1.png) ![Figure 19](image2.png)

**Figure 18** Representation of a scene with two points and an external mask. The relevant thing here is the value of the aperture that in this case is $\alpha_1$.

**Figure 19** Representation of the same scene of the previous image but $\alpha_1 < \alpha_2$. We want to emphasize the fact that for a bigger value of $\alpha$, less vignetting we get.
Intuitively, it seems clear that for bigger value of $\alpha$, more information will contain our sensor and less vignetting we will get. Now, we are going to prove that mathematically.

In order to do so, we will find the cutoffs between the sensor and the maximum ray light that we get. These cutoffs are the points that are delimitating the information our sensor gets. We can see them on Figure 20.

With the purpose of finding the cutoffs, we have to know which the equations of this schuss are. In the section 3.1.1.1 Mathematical explanation of vignetting, we have found the equation of the schuss that passes through the middle of the aperture. So, we will redo the same procedure with the aim of finding the equations that are delimitating our vignetting. The value of the aperture is $\Delta$.

On the following images, we can see the representation of the light ray crossing the virtual mask in order to arrive in the sensor as well as on Figure 15. But in this case, this light rays, are the limit ones. On Figure 21, we have the lower limit, and on Figure 22, we have the upper limit.

For similarity of triangles [8], for the lower limit, we can rewrite it like this:

$$\frac{s - u}{m - \frac{\Delta}{2} - u} = \frac{d}{d - d_m}$$

And for the upper limit:

$$\frac{s - u}{m + \frac{\Delta}{2} - u} = \frac{d}{d - d_m}$$
Beginning with the lower limit, if now we operate in order to isolate our unknowns, we obtain the following result:

\[
\frac{s - u}{m - \frac{\Delta}{2} - u} = \frac{d}{d - d_m}
\]

\[(s - u)(d - d_m) = \left(m - \frac{\Delta}{2} - u\right)d\]

\[sd - sd_m - ud + ud_m = md - ud - \frac{\Delta}{2}d\]

\[sd - sd_m = -ud_m + md - \frac{\Delta}{2}d\]

\[s \left(d - d_m\right) = -ud_m + md - \frac{\Delta}{2}d\]

\[s = \frac{-d_m}{d - d_m}u + \left(m - \frac{\Delta}{2}\right)\left(\frac{d}{d - d_m}\right)\]

\[s = \frac{-d_m}{d - d_m}u + \left(m - \frac{\Delta}{2}\right)\left(\frac{d + d_m - d_m}{d - d_m}\right)\]

\[s = \frac{-d_m}{d - d_m}u + \left(m - \frac{\Delta}{2}\right)\left(\frac{d - d_m}{d - d_m} + \frac{d_m}{d - d_m}\right)\]

\[s = \frac{-d_m}{d - d_m}u + \left(m - \frac{\Delta}{2}\right)\left(1 + \frac{d_m}{d - d_m}\right)\]

where \(\rho = \frac{d_m}{d - d_m}\)

\[s = -\rho u + \left(m - \frac{\Delta}{2}\right)(1 + \rho)\]
And for the upper limit:

\[
\frac{s - u}{m + \frac{\Delta}{2} - u} = \frac{d}{d - d_m}
\]

\[(s - u)(d - d_m) = \left(m + \frac{\Delta}{2} - u\right)d\]

\[sd - sd_m - ud + ud_m = md - ud + \frac{\Delta}{2}d\]

\[sd - sd_m = -ud_m + md + \frac{\Delta}{2}d\]

\[s(d - d_m) = -ud_m + md + \frac{\Delta}{2}d\]

\[s = -\frac{d_m}{d - d_m}u + (m + \frac{\Delta}{2})(\frac{d}{d - d_m})\]

\[s = -\frac{d_m}{d - d_m}u + (m + \frac{\Delta}{2})(\frac{d + d_m - d_m}{d - d_m})\]

\[s = -\frac{d_m}{d - d_m}u + (m + \frac{\Delta}{2})(\frac{d - d_m + \frac{d_m}{d - d_m}}{d - d_m})\]

\[s = -\frac{d_m}{d - d_m}u + (m + \frac{\Delta}{2})\left(1 + \frac{d_m}{d - d_m}\right)\]

where \(\rho = \frac{d_m}{d - d_m}\)

\[s = -\rho u + +\left(m + \frac{\Delta}{2}\right)(1 + \rho)\]
Now, we have all the equations we need to find the cutoffs. We have represented it on Figure 23 for easy reading. On this image, we also see the horizontal equations: \( s = S \) and \( s = -S \) where \( 2S \) is the maximum aperture of the lens.

![Diagram](image)

**Figure 23** We have illustrate all the equations that we need and the location where they are, in order to find the cutoffs

The aim is to find at which \( u \) the lines \( s = S \) and \( s = - \rho u + (m - \frac{\Delta}{2}) (1 + \rho) \) intercept, as well as the lines \( s = -S \) and \( s = - \rho u + (m + \frac{\Delta}{2}) (1 + \rho) \).

**Equation one** \( \rightarrow s = S \)

**Equation two** \( \rightarrow s = - \rho u + \left(m - \frac{\Delta}{2}\right) (1 + \rho) \)

\[
S = - \rho u + \left(m - \frac{\Delta}{2}\right) (1 + \rho) \\
\rho u = -S + \left(m - \frac{\Delta}{2}\right) (1 + \rho)
\]

\[
S + \rho u = \left(m - \frac{\Delta}{2}\right) (1 + \rho) \\
u_A = \frac{-S + \left(m - \frac{\Delta}{2}\right) (1 + \rho)}{\rho}
\]

And we do the same for finding the cutoff that is on the bottom, which we will call \( u_B \):

**Equation one** \( \rightarrow s = -S \)

**Equation two** \( \rightarrow s = - \rho u + \left(m + \frac{\Delta}{2}\right) (1 + \rho) \)

\[
-S = - \rho u + \left(m + \frac{\Delta}{2}\right) (1 + \rho) \\
\rho u = S + \left(m + \frac{\Delta}{2}\right) (1 + \rho)
\]

\[
-S + \rho u = \left(m + \frac{\Delta}{2}\right) (1 + \rho) \\
u_B = \frac{S + \left(m + \frac{\Delta}{2}\right) (1 + \rho)}{\rho}
\]
Now that we have the two cutoffs, we can prove that for bigger value of α we will have less vigneted. We have to remember that $\alpha = \Delta \sqrt{\frac{(1+\rho)^2}{1+\rho^2}}$. $\Delta$ is the size of the aperture and $\sqrt{\frac{(1+\rho)^2}{1+\rho^2}}$ are fixed values of the camera. So, if we increase the size of $\Delta$, the value of $u_A$ becomes more negative and has a result we are displacing this cutoff to the left. On the same time, the value of $u_B$ becomes more positive and as a result we are displacing this cutoff to the right. If we observe the following images, it is clearly that when we have a bigger aperture, our sensor has more information because we have more range.

![Figure 24](image1.png) **Figure 24** We can see the example that if we decrees the value of $\Delta$ We are directly decreasing the value of $\alpha$. Then cutoff $u_A$ increases and $u_B$ decreases

![Figure 25](image2.png) **Figure 25** We can see the example that if we increase the value of $\Delta$ We are directly increasing the value of $\alpha$. Then cutoff $u_A$ decreases and $u_B$ increases

### 3.1.2.2 Perspective

In this section, we will demonstrate mathematically the relation perspective has with the aperture of the external mask hole. First of all, we will explain it in an intuitive way. If we pay attention on **Figure 26 and 27**, we can see the same scenario of the previous images: two points and an external mask. What is interesting here is the coordinate of $I(u)$. $I(u)$ is the image that the sensor registers, where the height is representing the amount of light that comes to it, while the width is representing the distribution of the light among pixels.

In the case of the blue point, the distribution of this information that is arriving to the sensor, it always will be on the same point. Because it is on focus therefore, when we change the location of the hole although we are changing the perspective, remains on the same point. We can check that on **Figure 26** and on **Figure 27** where we do not have the position of the mask in the same place – the red lines are in a different position-, and without changing the location of the green and blue points, the distribution of the blue pixels on the sensor are in the same area.

On the other hand, as we have seen previously, the green point is not on focus. We can see that when we change the mask location, on **Figure 26**, the distributions of the pixels are centered on 2.5, whereas on **Figure 27** it is around 2. That is exactly what gives us
the sense of perspective change. On one image the green point is located on one place and on the other one it is not exactly on the same place as before. Moreover, as we have said, the height of $I(u)$ is representing the amount of information or the amount of light rays the sensor gets. Hence Figure 26 we can see that the height of the blue point respect to Figure 27, it is slightly lower because for the position of the mask does not cover as much as on Figure 26. However, for the case of the green point, the height remains equal because the information that the sensor is receiving is not changing.

We also can check that on Figure 28. If we pay attention carefully, we can appreciate that the red pen, have been moved ups and the green ones, down. In other words, exhibit a vertical parallax. However the yellow pencil, has remains on the same place, and that is because this pencil, is on focus and we cannot appreciate and different perspective between both images.

![Figure 26](image1)

*Figure 26* For the location of the external mask, we can see that the green point of the scene, the amount of light that is arriving to the sensor it is a value of 0.9, and its image location it is between 2 and 3. If we speak of the blue point, the amount of light we get is 1.6 and its location it is between 3.8 and 4. The scale value it is just a way to visualize better the change of location and the light information that we get, when we change the position of the external mask.

![Figure 27](image2)

*Figure 27* For the location of the external mask, we can see that the green point of the scene, the amount of light that is arriving to the sensor it is a value of 0.9, and its image location it is between 1.8 and 2.5. If we speak of the blue point, the amount of light we get is 1.3 and its location its is between 3.8 and 4. The scale value it is just a way to visualize better the change of location and the light information that we get, when we change the position of the external mask.

![Figure 28](image3)

*Figure 28* we can see a pair of images that exhibit vertical parallax. It we take a look on the red pens, we can see that from on image to the other, they vertical location has changed. The same is happening with the green ones. Source: Article of Light Field Photography with a Hand-held Plenoptic Camera

---

Once we have understood the mathematical concept of perspective, we will explain the relationship that the size of the aperture has with perspective.

On Figure 29 and 30, we have represented one point colored by orange out of focus – remember that if it was on focus it will be meaningless to speak of perspective-, and two masks.

On the first case, we have the green mask and the correspondent image that is arriving to the sensor. Then, on Figure 30 we have move these green mask upwards, and we get also the same image but in another place of the sensor. Therefore we have the sense of perspective, because on Figure 29 we are seeing the orange point in one place whereas on Figure 30, we are seeing the same orange point but in another part of the scene. The same is happening with the blue mask.

The aperture of the green mask is bigger than the blue mask. Hence, we can perceive that on the case of the green mask, the different location of the object is lower than on the blue case. In other words, the distance between the images acquired using the green mask is lower that the distance between the images acquired using the blue mask.

That means that for bigger aperture we will perceive less the perspective between images, because the distance will be lower. On the contrary, for smaller aperture, the distance of the location of the orange object between the images acquired with the different apertures will be bigger and that will means to have more sense of perspective because the location of the object for one image to another will change more.

In order to prove that mathematically, we will calculate the distance between both objects. In other words, we will compare which object differs more from the initial position when we change the hole of the mask.

To do so, we will calculate the points of intersection between the object out of focus and the mask. We can see more clearly that on Figure 31. On this image, we have join Figures 29 and 30. So we have represented the green and blue mask on the first position and its representation of the orange object on the sensor. Moreover we have both masks on the second position and also its representation of the orange object on the sensor. We have painted more blurred the green mask for an easy lecture.
As we have already seen in previous sections the equation that describes upper limit of the mask it is:

\[ s = -\rho u + \left( m_1 + \frac{\Delta}{2} \right)(1 + \rho) \]

And the equation that describes the point out of focus it could be anything; therefore, we have assigned the generic equation of line:

\[ s = au + b \]

Hence, in order to find the point of intersection, we just need to match these equations and operate:

\[ s = -\rho u + \left( m_1 + \frac{\Delta}{2} \right)(1 + \rho) \]

\[ au + b = -\rho u + \left( m_1 + \frac{\Delta}{2} \right)(1 + \rho) \]

\[ au + \rho u = -b + \left( m_1 + \frac{\Delta}{2} \right)(1 + \rho) \]

\[ u(a + \rho) = -b + \left( m_1 + \frac{\Delta}{2} \right)(1 + \rho) \]

\[ u_1 = \frac{-b + \left( m_1 + \frac{\Delta}{2} \right)(1 + \rho)}{(a + \rho)} \]
Then, we just need to repeat the same procedure for finding the second point of interaction. Newly the equation of the object has to be the same, because we do not have change the scene:

\[ s = au + b \]

And as we have already seen, the equation of the lower limit of the mask it is the following one:

\[ s = -\rho u + \left( m_2 - \frac{\Delta}{2} \right)(1 + \rho) \]

So we match again both equations and operate:

\[ au + b = -\rho u + \left( m_2 - \frac{\Delta}{2} \right)(1 + \rho) \]

\[ au + \rho u = -b + \left( m_2 - \frac{\Delta}{2} \right)(1 + \rho) \]

\[ u(a + \rho) = -b + \left( m_2 - \frac{\Delta}{2} \right)(1 + \rho) \]

\[ u_2 = \frac{-b + \left( m_2 - \frac{\Delta}{2} \right)(1 + \rho)}{(a + \rho)} \]

Once we have both points of intersection, we could know how much the object have changed its perspective, by knowing the distance between the first position and the second one. So, we just need to make a rest of the two points of intersection funded before:

\[ |u_1 - u_2| = \left| \left( \frac{-b + \left( m_1 - \frac{\Delta}{2} \right)(1 + \rho)}{(a + \rho)} \right) - \left( \frac{-b + \left( m_2 - \frac{\Delta}{2} \right)(1 + \rho)}{(a + \rho)} \right) \right| = \]

\[ = \left| (-b + m_1 + m_1 \rho + \frac{\Delta}{2} + \frac{\Delta}{2} \rho - (-b + m_2 + m_2 \rho - \frac{\Delta}{2} - \frac{\Delta}{2} \rho)) \frac{1}{(a + \rho)} \right| = \]
Now, we have the equation of the sense of perspective and we just have to analyze it. As we have explain before, \( m_1 \) and \( m_2 \) are the central position of the external mask. Where \( m_1 \) is for the first mask and \( m_2 \) is for the second. The value of \( \alpha \) is the slope of the object and \( \rho \) is the relation of distance between the mask the lens and the sensor. So, all those things are fixed values. The only things that will vary our perspective it is the value of the aperture: \( \Delta \).

As in our equation we have an absolute value, we have to be careful in study if the numerator and denominator, are positive or negative. We will start by studying the denominator. As we have said, the value of \( \alpha \) is a slope that could be positive or negative. On the other hand, and \( \rho \) is the relation of distance so it could not be negative. Anyway, the sign of \( \alpha \) will not affect on the absolute value.

Now let’s focus on the numerator. We have to study if it is positive or negative. As we have explained, \( m_1 \) is the central position of the first mask and \( m_2 \) the central position of the second one. Therefore, \( m_1 < m_2 \), what makes that \( \Delta < m_1 - m_2 \) will always be negative. As we have said, \( \rho \) could not be negative. So we will have to study the different values of \( \Delta \) that makes our numerator positive or negative and we find three cases:

- \( \Delta = m_1 - m_2 \)
- \( \Delta > m_1 - m_2 \)
- \( \Delta < m_1 - m_2 \)

If we are in the first case: \( \Delta = m_1 - m_2 \) the numerator will be 0 and the distance will be zero, so we will not have sense of perspective. In an intuitive way, if we have the both mask on the same center of position it is evident that we will not generate perspective. On the second case, \( \Delta > m_1 - m_2 \) the numerator will be positive. Hence if we remember the distance equation \( \frac{|\Delta|(m_1-m_2)(1+\rho)}{|(\alpha+\rho)|} \) for bigger values of \( \Delta \), we will have bigger distance, which will not make sense in our intuitive explanation. But if we pay attention if the value of \( \Delta \) is bigger than \( m_1 - m_2 \) it means that \( m_1 > m_2 \), therefore the masks are overlapping or simply are crossing. This fact will represents that for the position of the holes, we are getting images that are sharing the same light ray directions. So as a result, we would not be isolating well. Neither the first case neither the second case, make sense to us because we are using different holes on the mask with the aim of having an array of images with different
light ray direction. Hence we can reject this cases because they will not going to happen.

So, let’s focus on the last and interesting case: $\Delta < m_1 - m_2$. In this case, the numerator will be always negative. According to the absolute value function, if we are on the negative part – which when $\Delta < m_1 - m_2$ we are always in this part - , for bigger values of $\Delta$ the function decreases and gets close to 0. So for bigger values of $\Delta$, the function will decrease, and as a result the distance will decrease too. This reasoning matches with our intuitive explanation. As much as the value of the aperture increases, the distances will decreases resulting a loss of less sense of perspective.

![Graphical behavior of absolute value of $\Delta$](image)

**Figure 32** Graphical behavior of absolute value of $\Delta$

3.1.3 Geometrical distance between limit light rays

On this section, we are going to prove mathematically the geometrical distance between the limit light rays. We have called this distance: $\alpha$. On section 3.1.2.1 Vignetting, we have already prove that the mathematical equation that defines the upper light limit and the lower light limit are:

Upper limit: $s = -\rho u + \left( m + \frac{\Delta}{2} \right) (1 + \rho)$

Lower limit: $s = -\rho u + \left( m - \frac{\Delta}{2} \right) (1 + \rho)$
On the following image, we have all this concepts represented in order to see it clearly:

![Diagram](image)

**Figure 33** we can see the equation for the limit light rays and the equation of the middle light ray.

Now, we want to know which the value of is $α$. Hence, we have to compute the distance between two parallel lines, which we will define as ‘$r$’ and ‘$t$’:

$$r \equiv s = -ρu + \left(m + \frac{Δ}{2}\right)(1 + ρ)$$

Then we equalize the schuss to zero:

$$r \equiv 0 = -s - ρu + \left(m + \frac{Δ}{2}\right)(1 + ρ)$$

And we do the same procedure for the other schuss:

$$t \equiv s = -ρu + \left(m - \frac{Δ}{2}\right)(1 + ρ)$$

$$t \equiv 0 = -s - ρu + \left(m - \frac{Δ}{2}\right)(1 + ρ)$$

The following formula, computes the distance between parallel lines:

$$d(r, t) = \frac{|C - C'|}{\sqrt{A^2 + B^2}}$$

Where $A, B, C$ and $C'$ are:

$$r \equiv Ax + By + C = 0$$

$$t \equiv Ax + By + C' = 0$$
We have our coordinate system as it is illustrated on the following image:

![Figure 34 Conventional coordinate system](image1)

![Figure 35 Coordinate system we are working with](image2)

So, we can identify easily the values of A, B, C and C’:

\[
A = -\rho \\
B = -1 \\
C = \left( m + \frac{\Delta}{2} \right)(1 + \rho) \\
C' = \left( m - \frac{\Delta}{2} \right)(1 + \rho)
\]

Then we just need to replace A, B, C and C’, for values that we have in this case on the equation:

\[
d (r, t) = \frac{| \left( m + \frac{\Delta}{2} \right)(1 + \rho) - \left( m - \frac{\Delta}{2} \right)(1 + \rho) |}{\sqrt{(-\rho)^2 + (-1)^2}} = \\
d (r, t) = \frac{| \frac{\Delta}{2} + \frac{\Delta}{2} \rho - \left( -\frac{\Delta}{2} - \frac{\Delta}{2} \rho \right) |}{\sqrt{\rho^2 + 1^2}} = \\
d (r, t) = \frac{| \Delta + \Delta \rho |}{\sqrt{\rho^2 + 1^2}} = 
\]
We can remove the absolute value because \( \Delta \) and \( \rho \) will always be positive, hence \( \Delta + \Delta \rho \) must be positive too:

\[
\begin{align*}
  d(r, t) &= \frac{\Delta + \Delta \rho}{\sqrt{\rho^2 + 1}} = \\
  d(r, t) &= \frac{\Delta(1 + \rho)}{\sqrt{\rho^2 + 1}} = \\
  \alpha &= d(r, t) = \Delta \left( \frac{(1 + \rho)^2}{\sqrt{\rho^2 + 1}} \right)
\end{align*}
\]

### 3.2 Building a lightfield image

In this section it is explain the different designs for the external mask that have been implemented. It also explains the appropriate parameters that we have to adjust to our camera.

#### 3.2.1 Design of the external mask

As I have explained previously, plenoptical cameras contain an array of micro-lenses placed between the main lenses and the sensor. Having this array, we keep the light direction information, as well as the intensity of light. As we are not going to change the internal parameters of the camera, we have implemented and external mask that has allowed achieving different light directions.

The initial approximation was the construction of a geometrical cube, with 5x5x5 cm dimensions. We have measured the diameter of the objective of the camera – that was five centimeters- in order to place the objective inside the cube. As the length of the cube was the diameter of the objective, when the objective was placed inside, the mask has had to be fixed. We can see the first test on Figure 36.

After doing this it was required to cover the objective with a black card with a hole. This way, we are only allowing passing through the camera a specific light direction. As I was focused on the mask construction, the dimension of the holes, were random – later we will focus on the properly size and shape-. We can see set of this cards on Figure 37.
When we started to shoot for checking the correct functionality, we realized that the model was not optimal. Unfortunately, the geometrical cube was not a good implementation because of two reasons. The main one was that if the objective was inside the mask, we were not able to make focus in a manually way, because the camera that we were using has this parameter at the end of the objective. The second one was that, as the length of the mask was 5 cm, we were putting the cover – the black card – five centimeters farther from the objective. So we realized that we were generating much more vignetting. As a result the hole must be bigger and we were decreasing the sense of having the same scene with different perspective, as we have explained on section 3.1.2.2 Perspective.
After realizing that this model was not giving the expected results, we decided to implement a new prototype. Analyzing the problems that the first test had, will give us a better result. Hence, for the second test, we will try to put the mask so close to the optical center in order to avoid as much as we can vignetting. As a result, the mask will be located right in front of the objective. Furthermore, the new design must allow to change the parameter of focalization of the camera in a manually way. With the previous model, the part of the objective of the camera that allows us to regulate the focalization was inside the cube, so as a result, we could not access to this regulation. Therefore, we have implemented a thin sheet with a square hole of the measurement of the objective on the center. We can see it on Figure 40.

![Figure 40 Thin sheet for the implementation of the external mask](image1)

![Figure 41 Camera with the thin sheet for the implementation of the external mask](image2)

Unfortunately, this model was not also good. Although it was a good design in terms of avoiding extra vignetting because of the mask was located right in front of the objective, the thin sheet has problems with blocking light. As the mask was not stuck to the objective, behind the external mask came a small beam of light. The goal of the external mask is to act as a filter that only allows passing a single light direction. So it does not have sense to use this external mask because the sensor is receiving a concrete light direction in addition to the back light.

Finally, we implemented the optimal external mask. The external mask requires basically two things: that the mask is placed right in front of the objective and that the only light that arrives to the sensor is the one that the mask allows passing. So we realize that it was required to buy something that stuck the thin sheet that we talked before with the objective. So, we have bought a filter. We can see it on Figure 42.

![Figure 42 Camera with the thin sheet for the implementation of the external mask](image3)

As we only need the black ring, we have broken the lens for avoiding optical distortions. Then we have paste the ring with the thin sheet. As a result, we have create an external mask which is acting has a perfect filter of light directions and moreover, this design is allowing us to regulate the parameter of focalization manually. We can see the final result of the external mask on Figure 43.
3.2.2 Shape, size and position of the holes

3.2.2.1 Shape

Once we already have the optimal design for the external mask, we have to think for the optimal shape, aperture and position of the hole that have to be on the black cards. Apparently, the shape of the hole it is not something so much relevant, so I have decided that the shape will be round. However the decision of the aperture and position it is something very critical.

3.2.2.2 Size

It has been so difficult to achieve the engagement between vignetting and perspective. When we put the external mask, we are generating vignetting as a result of the occlusion of many rays. However, if our scene is really vigneted it will be impossible to process the image, hence we could not afford many vigneted in our image. For the other hand, the aim of the external mask, is to generate some images of the same scene where the camera is always static where it seems that all these images have been shooted in different perspectives. The problem begins when: bigger we make the holes, less vigneted we will get. As it is expected, smaller we make the hole, more vigneted we will get. Nevertheless, bigger we make the hole, less sense of changing the perspective we get and vice versa. We can check it mathematically on 3.1.2Mathematical demonstration between the engagement vignetting and perspective.

In order to determinate the size of the hole we have to look after the zoom and the number f of the camera we are using. We will explain that on section 3.2.3 Camera preparation. But for the parameters of the camera that we have, which are a zoom of 55 cm and a number f of 5.6, after have tried many different sizes, we have realized that the properly once was having a circle hole of 0.1 mm of diameter. As the zoom is big and the aperture is the biggest once with this zoom, we do not need that the hole have a

Figure 42 Filter of 52 mm. We will use it in order to block all the ray lights and only allow to pass the light direction that we desire

Figure 43 Final design of the external mask. It is composed by the thin sheet pasted on the filter. This way we do not have the problem of letting pass the back light through the camera.
diameter of 0.5 mm or something bigger, because we are close to the lens. Hence with this little hole, we obtain an image that does not contain many vignetting. This is an advantage because as much little we could make the holes; much distance will exist between them. Here we can appreciate a pair of images which have been taken with an external mask which each hole is a circle of 0.1mm. As we can appreciate, we can afford this vignetting that we obtain with this hole size because there is not many.

Figure 44 Examples of a pair of images which have been shoted with a zoom of 55mm and a number f of 5.6. Also as we could appreciate some vignetting in front of the camera we have put an external mask. The cover contains a hole of a circle of 0.1 mm of diameter

In the following images, we can see a pair of examples of cases that we have been searching for the right hole size. In all this cases, the parameters of the camera were the same – we will explain that in the section below 3.2.3 Camera preparation. On the first case, Figure 45 the size of the hole was 0.5 mm. This size was fantastic in terms of vignetting. We can hardly find vignetting on our images; however, as we have said before, we have to find an engagement between vignetting and perspective. It is really difficult to find any difference of perspective between these two images. This is because the hole it very big and as a consequence these two images are sharing many light rays.
These images have been taken with an external mask that has an aperture of 0.5mm. We can see that we could not appreciate any change of perspective.

On the other hand, on *Figure 46* we can see a pair of images which the hole of the cover is 0.1mm. We can appreciate that compared with the images above we have more vignetting in our images. However, the amount of information that is arriving to the sensor it is acceptable. Unlike to the other images, here we can see a different change of perspective. On the image below the objects have been move slightly to the right. We have circled some objects in order to focus the attention on the places that this change of perspective is easier to see.
3.2.2.3 Position

Besides, we also have to take care with the position of the holes, because if the holes are too close, we will lose the sense of perspective because the difference will be really small. Moreover, we have to be careful not to overlap any hole. Overlapping will cause that our images will share light ray directions so they will have the same perspective. We have to remember that we are creating the external mask in order to block rays in order to have a single direction for each image; hence it will be meaningless to overlap the holes.

With the external mask, as we have already said, it is used for achieving some images with different perspectives. In our case, we want to generate nine images. The disposition of the holes will be like a matrix of 3x3. As we have said on the section above the aperture of the camera was a circumference of 2 cm of diameter so with a size of 0.1 cm for each hole, we have located all this nine holes like it is illustrated on Figure 47.
The initial idea was to have a set of black cards where each card has a single hole. In other words if we desire to have nine images, we have create nine cards where each card have a single holes of the nine once. We can see it on the Figure 48.

After have shoted many times the scene with this cards, we realized that we could make this more optimal in order to reduce the human error. If we have to make nine holes in different cards, it is difficult that we have the precision of located all the holes in the exactly place that they have to be, because we are doing it manually. For example, when we are cutting the paper for make the hole, unconsciously maybe we are overlapping a little bit with the neighbor hole of the other card. Moreover, every time we put the card in the thin sheet, we do not have the precision of putting all the cards in the same place, so again, maybe we are overlapping holes, or reducing the distance that we desire that holes has. So we arrive to the conclusion that we were accumulating many human errors.

As a result, in order to prevent that, we have created a single card which will contain all the holes and we will stick it to the thin sheet. We can see it on the Figure 49. This way, we are sure that all the holes are located right where we desire and we do not have problems in changing the cards because all the times is the same card; which is fixed. The only thing that we have to take care in making the cover like this is to make sure that we are covering properly the rest of eight holes that we do not desire that the light rays passes through them.

Figure 47 Display of the holes for the cover of the external mask
3.2.3 Camera preparation

Now we already have the implementation of the external mask, hence we can start to shoot the scene. But first of all, we have to take care of some parameters of the camera and adjust them. Now we are going to explain the procedure that we have followed. It is really important that all the images that we have taken have the same parameters. In other words, that all of them has the same focus point, light and aperture. As we are putting and external mask, if we have the camera in automatic mode, for each different mask, the camera is going to change some parameters, so the first step have been to put the camera in manual mode. With this mode, the camera will only change the parameters that we desire.

Once we have the camera in manual mode, the next step has been to find one point on the scene where the camera will be on focus. It is relevant that we chose properly the focus point. When later we want to process the images, we need that the camera is focused at the bottom. For example if it is a scene like the example on Figure 48, we must make focus on the wall. However if the scene is outdoors, we must make focus on the infinity. This is because we will compute the change of disparity by computing the optical flow between images. This algorithm detects more movement farther of focus we are. Contrary will detect less movement as closer to the focus we are. Hence if we make focus at the bottom the objects that are located at the bottom its movement will be lower and viceversa.

Moreover, we have decided to use a zoom of 55 millimeters because the objective is more close to the lens and as a result, we can make smaller holes. Hence, if the holes have a smaller size, the distance between them are bigger. This way, as the holes are farther between them, the resulting sub-images presents a remarkable disparity.
Besides, we have to open as much as we can the aperture of the diaphragm. This is because we desire that the only thing that blocks the light rays be the external mask. Hence, the minimum number $f$ with this zoom, it is $5.6$. Another important point has been the shutter speed. All the images have to share the same illumination. In order to have all the images with a homogeneous illumination I have put the external mask whit the central hole and I have regulated it. The shutter speed depends on the size hole and how illuminated is the scene, but usually with the size of 0.1 mm, the exposition time, have been around 20 or 30 seconds.

The reason to choose the central hole is because if I adjust the light with one hole that it is located on one extreme, when we take the image with the hole that is located on the opposite extreme, the illumination is changing a lot. This way, we have obtained all the images well illuminated. Finally, we have put the value of 100 for the ISO, because the SNR is much higher for lower values of ISO.

Now, it is all ready and we can start shutting the scene in order to get all the images. Here we can see a simplification of 4 images, in order to check that all the procedure was working as we expected. This simplification has served to ensure that the hole size and the location were properly in order to satisfy the engagement between vignetting and perspective. Moreover, it served to test that the external mask was blocking the light rays as it should and if the parameters of the camera.

![Figure 50](image)

In this case, the external mask was prepared in order to generate 4 images. With the case of 9 images, we will achieve more accurate results. This is just a simplification in order to test the properly functionality of the external mask.
3.3 Estimating depth

Once we have verified that the trade-off between vignetting and perspective is fine, all sub-images share the same light and have been captured with the same camera parameters, we can start processing all the images. In our case we are working with the scene showed bellow. This image, have been shoted without any eternal mask. As we can appreciate, we have put the focus at the bottom.

Our desire, with the help of the external mask, is to catch nine images of this scene, where the display of the holes will be like a 3x3 matrix, in order to obtain a depth map of the scene. With the depth map, as the name suggest, we will know which object is closer to the object that is in focus.

Figure 51 General image of the scene that we will work on. This image, have been shoted with a zoom of 55 mm and a number f of 5.6 in order to achieve the maxim aperture

After taking a photography using each hole, we have obtained the following array of nine images. As we expected putting the external mask, we can appreciate that in our images in some areas the sensor is not having any type of information and as a result we have vignetting. On the other hand, if we pay attention, we can see that we are achieving this change of perspective that we have talked before. It is a little hard to see it because the image size is little and the change of perspective it is not really big. However If we pay attention to the vertices of the first pyramid, we can clearly see that depending on the sub-image is located in a different place.
The aim of processing all this nine images, as we have already said, is to obtain information about the depth of the scene. In other words, we want to achieve a depth map where we could see which objects are closer to the object in focus.

In order to do so, we have divided the explanation of this development in three parts: data initialization, optical flow calculation and data processing. We have processed our images and as a result created the depth map using Matlab for the calculations.
3.3.1 Data initialization

Data initialization is the first part of this block diagram. We can detach this block in four blocks more: read images, mask creation, erase vignetting and images relation.

![Block diagram](image)

**Figure 54** Block diagram that summarizes the first part of the general block diagram

When we already have the nine images, and we have read it, we are ready to begin. First of all, we create the mask $\chi: \Omega \rightarrow [0,1]$ modeling the vignetted mask, so that $\chi(x,y) = 1$ identifies the non-vignetted pixels. On *Figure 55*, you can see the example, where white color means $\chi = 1$ and $\chi = 0$ means the black color.

As $\chi$ is identifying the non-vignetted pixels, we need a $\chi$ mask for each of the nine sub-images. In order to obtain the adequate $\chi$ for each sub-image, we have put the external mask that we have created to the camera. Then we have put a light bulb in front of the camera and a shutter speed of 30 seconds.

Our desire is to saturate the sensor. This way, we will obtain an image where the part that the sensor has no information will be black; however, the non-vignetted pixels will be white. In other words, we will obtain an image that all this pixels are black in the same location where our sub-image has vignetting, and will be white in the same location that our sub-image is getting information of the scene. Then we just have to quantize of image, in order to obtain a binary one and repeat this process for all the holes of the external mask.

This step is really important because, when we desire to delete the vignetting in order to calculate the optical flow, the mask will act has a reference. In other words, this $\chi$ mask will say us, where on our sub-images, we have pixels whit right information, when the corresponding $\chi$ is $\chi = 1$, and which pixels are wrong, when the corresponding $\chi$ is $\chi = 0$. 


On the left we can see one sub-image of the nine array we have achieved with the use of the external mask. On the right, we can see the corresponding mask that determines on with the pixel information that will be useful for our image processing, and on black all the pixels that we have to reject.

Having the nine different $\chi$ masks for the nine sub-images, we can start the process of deleting vignetting.

As we have said, by computing the optical flow between two images, we will obtain the change of disparity. We will talk deeper of this calculation on the next block diagram called optical flow calculation.

At the beginning we have not delete any vignetting on our images and when we desire to calculate the optical flow, with this vignetting the algorithm gets really lost. This is because vignetting is a uniform surface, then having not a texture the algorithm, was not detecting the pixels properly. Hence we have proceeded to erase it and moreover replace it for a random surface which will create a texture that will help the algorithm for the well detection.

In order to do so, we have implemented a function which its aim is to return two images where we can appreciate the change of perspective that has suffered the images without any vignetting. As the positions of the holes of the external mask are displays as a 3x3 matrix, we have called our nine sub-images in a matricial way: $I_{1,1}$, $I_{1,2}$, $I_{1,3}$, $I_{2,1}$, $I_{2,2}$, $I_{2,3}$, $I_{3,1}$ $I_{3,2}$, $I_{3,3}$.

If we enter for example $I_{1,1}$ and $I_{1,2}$, this function will return two images where, on the places where both images have vignetting, hence $\chi_{1,1}^{(i,j)} = 0$ and $\chi_{1,2}^{(i,j)} = 0$, we will assign the same random value on $I_{1,1}^{(i,j)}$ and $I_{1,2}^{(i,j)}$. The reason to put the same random value on the same pixels in both images is because we want to generate the same texture for both images. This way, when we will compute the optical flow, for the algorithm will be easier to make well the pixel correspondences.

Besides, as we desire to eliminate all the vignetting in both images, we have to look after of two more cases.

The first one, we have to take care when $\chi_{1,1}^{(i,j)} = 0$ however, $\chi_{1,2}^{(i,j)} = 1$. In other words $I_{1,1}^{(i,j)}$ have vignetting but $I_{1,2}^{(i,j)}$ not. We will copy the pixel value of $I_{1,2}^{(i,j)}$ - where we do not have vignetting- and we will replace it on our $I_{1,1}^{(i,j)}$. 

\[\text{Figure 55} \quad \text{On the left we can see one sub-image of the nine array we have achieved with the use of the external mask. On the right, we can see the corresponding mask that determines on with the pixel information that will be useful for our image processing, and on black all the pixels that we have to reject.}\]
The second case is the same but upside-down of the previous case: \( \chi_{1,1}^{(i,j)} = 1 \) however, \( \chi_{1,2}^{(i,j)} = 0 \), then as a result we will change the pixel value \((i, j)\) of \( I_{1,2} \) by \( I_{1,2}^{(i,j)} = I_{1,1}^{(i,j)} \).

We have detected that the computed optical flow on the vignetted areas negatively affects to the computation of the overall optical flow. This is because of mainly two reasons; first, there is a region in which the values are constant black. Second, there is a region that appears in one image, and not in the other - and vice versa. The first reason affects in the sense there are multiple choices of the optical flow for the pixels on homogeneous regions, and the second reason makes that there are pixels without any real correspondence between the two images, obtaining erroneous correspondence - optical flow -. Due to the regularization condition, this bad optical flow affects to the rest of pixels. In order to overcome this behavior, we replace the vignetted pixels with the same random values when the vignetted region is the same on the two images and with the values from the other image when the vignetted area affects only to an image.

On the top following images we can see the \( I_{1,2} \) on the left and \( I_{1,3} \) on the right after have being processed. On the bottom, we can appreciate these same images, once we have processed them. As we can appreciate there is not any vignetting on them.

![Figure 56](image-url) On the top we can see two sub-image of the nine array we have achieved with the use of the external mask. On the bottom, we can see the corresponding images where we have removed all the vignetting. On the black pixels we have assigned a random value if both sub-images have vignetting or we have assigned the value of the other sub-image in case that this sub-image has no vignetting.
3.3.2 Optical flow calculation

The second part of this general block diagram it is the optical flow calculation.

We have to remember that what does the optical flow [9], it is the calculation of the movement that the pixels have done to one consecutive frame to another. This computation is not something trivial, and doing this computation for images taken in a real ambience could be really complex. That is because the existing algorithm are based in hypothesis like intensity and contrast, object’s rigidity, among others, and generally not met in real scenarios.

So in order to compute the optical flow for our images, after trying many algorithms, we have decided to use TV-L1 Optical Flow Estimation algorithm. This method is based on the minimization of a functional containing a data term using the L1 norm and a regularization term using the total variation of the flow. The main feature of this formulation is that it allows discontinuities in the flow field, while being more robust to noise than the classical approach of Horn and Schunck. The algorithm is an efficient numerical scheme, which solves a relaxed version of the problem by alternate minimization [10].

So we have uploaded our pair of related images that we have computed on the previous block diagram to the available online demo\(^8\). When we have our images on the web, we have to decide which the suitable values are for \(N\) and \(\lambda\).

\(N\) it is the number of levels we desire. As this algorithm is computed as a Gaussian pyramid, with the two images we have, we downsample many times as the number of levels. Then on the greater level, we compute the optical flow and we go up on the next level, where images are interpolated and we take the optical flow we have computed as initialization and we compute again the optical flow. We have to do this downsample

\(^8\) The link of this available online demo it is the following one:  http://demo.ipol.im/demo/26/
for the assumption that we have little movements. It is because of that, that for greater N, we will be able to detect greater movements. The parameter N, have its limits between 1 and 8. For N equals to 1 the algorithm detects only sub-pixel movements. On the other hand, for N equals to 8, the algorithm detects a large displacements.

λ is the parameter that controls the smoothness of the optical flow, in other words the regularity of the solution. If we have two neighbor pixels that have opposite movements, if we are forcing λ to be smooth, this algorithm will not detect that this two pixels have opposite movements. It will detect that both movements are similar. On the other hand, if λ does not have to be smooth, it will be able to detect the opposite movements of the neighbor pixels, however, in the homogeneous areas of images, the algorithm will get lost and will not make well the pixel correspondence to one image to another.

The parameter of λ, have its limits between 0.01 and 1. For λ equals to 0.01 the algorithm forces a very smooth flow. On the other hand, for λ equals to 1, the algorithm allows a very discontinuous flow.

Finally, when we have got the properly parameters of the algorithm, we just have to download the optical flow that relates our two sub-images that we have updated.

3.3.3 Data processing

Data processing is the last part of achieving our depth map of the scene. This section is divided in five parts: Read flow files, mask relation, module calculation, convolution and pixel detection.

First of all, we read the flow file that we have achieved in the previous block diagram. In this next step, we have to relate the mask. As we have said, the only part of every image that is giving us information are the pixels where, its corresponding \( \chi_{XY}^{(i,j)} = 1 \). Hence, as each flow file is giving us the change of disparity between two images, we have to create a new \( \chi \) mask that suits properly for each flow file.

We have to remember that the change of disparity that each flow file is giving to us is change of \( I_{XY} \) is having between \( I_{X,Y} \) but only on the parts where \( \chi_{XY}^{(i,j)} = 1 \) and \( \chi_{X,Y}^{(i,j)} = 1 \). Hence, if the flow file is relating the images \( I_{XY} \) and \( I_{X,Y} \), the new \( \chi_{XY,Y} \) mask will be the sum of the respective masks of each image: \( \chi_{XY,Y} = \chi_{XY} + \chi_{X,Y} \).
The next step has been to compute the module of each flow file. This way, we are painting in gray scale which objects are deeper and which once are closer. To continue, we have convolved each image with a Gaussian. The reason why we have done so is because if we make this convolution with a Gaussian, we are making our image blurred. The last step of this process will be to do the average of all images. We have to contemplate that in our images all the objects are not in the same exact pixel location, so in order to reduce this error, we have decide to make blurred the image. This way the average will be better.

As we have said the last step consisted in doing the average between all images. But first of all, we have to make pixel detection in order to reject those pixels on all images $I_{xyx'y'}^{(i,j)}$ that are not giving us the right information, in other words the once that $\chi_{xyx'y'}^{(i,j)} = 0$.

So we have put in an auxiliary matrix, all the pixels values of all images $I_{xyx'y'}^{(i,j)}$ where each $\chi_{xyx'y'}^{(i,j)} = 1$. And then we have divided for the number of times we have summed in each $(i,j)$ location of the auxiliary matrix in order to do the average and obtain a depth map.
2.3.4 Results and limitations

The depth map we have obtained after having processed the nine sub-images we have showed on Figure 59, is the following one:

![Depth map of the scene](image1)

**Figure 59** Depth map of the scene we have represented on Figure 51

In this depth map we have obtained, we can see which objects of the scene are behind and which ones are ahead. Deeper we go in the scene is colored by black and the pixel value is around 0, because the focal point for this image was on the background.

In the following image, we have an aerial view of the scene. This way we can check if the representation of the depth map is correct.

![Aerial view of the scene](image2)

**Figure 60** Aerial view of the scene represented on Figure 51
In the following images we have illustrated the correspondences in order to check properly the results of the depth map:

Figure 61 Illustration that makes the correspondences of Figures 51, 59 and 60
Besides, we also show the pixel value of the three objects of the scene and also the value of the wall, in order to check it numerically:

**Figure 62** Depth map. We are highlighting the pixel value of the element that is closer. We can check that it is alright because its value is 7.5

**Figure 63** Depth map. We are highlighting the pixel value of the second element that is closer. We can check that it is alright because its value is 4.8

**Figure 64** Depth map. We are highlighting the pixel value of the third element that is closer. We can check that it is alright because its value is 2.671

**Figure 65** Depth map. We are highlighting the pixel value of the element that is at the bottom. We can check that it is alright because its value is 0.6

**Figure 66** Depth map. We are highlighting the pixel value of the wall. We can check that it is alright because its value is 0.2
Moreover, we attach other results that have been achieved properly. On Figure 67 we illustrate the new scene which we desire to get the depth map. On the Annex II. Lightfield images we can see all the nine sub-images we have recorded in order to process them and obtain the depth map we shown on Figure 68. We can clearly see that the little houses are ahead, the orange on the middle and the cans of lentils and beams are on the bottom. On the depth map we can easy visualize it, because of the little houses are painted in white, the orange is painted in yellow and the cans are mostly black.

**Figure 67** General image of the scene that we will work on. This image, have been shooted with a zoom of 55 mm and a number f of 5.6 in order to achieve the maxim aperture

**Figure 68** Depth map of the scene illustrated on Figure 67
In the following images we have illustrated the correspondences in order to check properly the results of the depth map:

Figure 69  Illustration that makes the correspondences of Figures 67 and 68 and the aerial view of the scene
Besides, we also show the pixel value of the three objects of the scene and also the value of the wall, in order to check it numerically:

- **Figure 70** Depth map. We are highlighting the pixel value of the element that is closer. We can check that it is alright because its value is 4.1.

- **Figure 71** Depth map. We are highlighting the pixel value of the second element that is closer. We can check that it is alright because its value is 3.7.

- **Figure 72** Depth map. We are highlighting the pixel value of the third element that is closer. We can check that it is alright because its value is 1.9.

- **Figure 73** Depth map. We are highlighting the pixel value of the fourth element that is closer. We can check that it is alright because its value is 0.8.

- **Figure 74** Depth map. We are highlighting the pixel value of the wall. We can check that it is alright because its value is 0.4.
Moreover, we desire to obtain a depth map of a scene which was outdoors. However the result has been disappointing. The most important part when we generate the depth map is the calculation of the optical flow. Therefore, we are really limited by the conditions of it.

In the scene illustrated on Figure 76, we expected to obtain a depth map where the chair closest to the camera was painted in white, then the table painted in yellow as well as the other chair, because they are more or less on the same location and finally the fences in black.

![General image of the scene that we will work on. This image, have been shoted with a zoom of 55 mm and a number f of 5.6 in order to achieve the maxim aperture](image1)

![Depth map of the scene illustrated on Figure 75](image2)
We have illustrated as well the correspondences in order to check the results of the depth map:

![Illustration of correspondences](image)

Figure 77  Illustration that makes the correspondences of Figures 75 and 76

The positive part is that as we can observe from Figure 78 to 81 that says us the exact values of some pixels, is that the depth map telling us some right information. The color that is representing the pixel value for the table, the second chair and the fences, is coherent. We can see that the fences is colored by black which indicates that is the farthest part of the scene, and then the table and the chair have the same yellow value.
which means that are located at the middle of the scene with a little distance between
them, which in fact, it is true.
However, we also see a clear incoherence in this depth map. The chair with should be
colored in white because it is the object more close to the camera, in fact it is colored
with red and black as if it was located near to the fence. This error is because the wrong
calculation of the optical flow. As we can see on Figure 75 does not have any texture.
When the algorithm is trying to make the pixel correspondences from one image to
another, in order to identify the movement direction is making it wrong taking the
nearest pixel has the pixel correspondences. As a result we get no movement.
On the other chair we have been luckier because although we have the same uniformity
on the chair, for the mounting of the scene, we have more pronounced edges which
helps the algorithm for the properly computation.

![Figure 78](image1.png)  Depth map. We are highlighting the pixel value of the
closest chair: 0.01

![Figure 79](image2.png)  Depth map. We are highlighting the pixel value of the
table 1.1

![Figure 80](image3.png)  Depth map. We are highlighting the pixel value of the
second chair: 0.9

![Figure 81](image4.png)  Depth map. We are highlighting the pixel value of the
fences 0.02

On the next image, we also desire to show other limitations where we have been stuck
for the conditions that the optical flow has. We can see a scene where the railing is on
focus and the houses and the landscape before it are out of focus. Hence, the depth map
that we expected to get is a depth map where the railing was completely black as
synonym of no movement and the other things colored from red to white as we move
away from the railing.
We found the main problem on the left part of the images because of the flux
discontinuity. The optical flow, has the assumption that the closest pixels, has more or
less the same movement. Having in main this assumption, and seeing that the background of this part have a similar color, as the railing has no texture, the pixels from one sub-image to another the algorithm has many options to determinate the pixel movement of the railing. As a result the algorithm determines that the railing is moving a lot.

The other problem is with the background. On the depth map, a part of the closest house, we could not identify anything. This is because all of them are so far away that the algorithm considers that are on the same distance. The houses are so far that have the same parallax. Besides they are relatively close between them and both so far of the focus point, the railing. Again for the limitations on regular field the optical flow has, it is close to impossible to obtain an accurate depth map for this kind of scenes.

Figure 82  General image of the scene that we will work on. This image, have been shoted with a zoom of 55 mm and a number f of 5.6 in order to achieve the maxim aperture

Figure 83  Depth map of the scene illustrated on Figure 82
On the following images we show some pixel values for contrasting what we have explained.

**Figure 84** Depth map. We are highlighting the pixel value of the railing 0.6

**Figure 85** Depth map. We are highlighting the pixel value of the closest house 2.1

**Figure 86** Depth map. We are highlighting the pixel value of the landscape 4.2

**Figure 87** Depth map. We are highlighting the pixel value of the sky 5.3
Chapter 4

CONCLUSIONS AND FUTURE WORK

In this work we have shown how we converted a conventional reflex camera in a plenoptic one. These types of cameras provide directional information on its images that is not present in common images. This directional information will allow solving some computer vision problems as depth estimation.

The article in which this project is based, “External mask based depth and light field camera” done by Dikpal Reddy, Jijamin Bai and Ravi Ramamoorthi, has a lack in the mathematical proofs of the optimal camera parameters that have to be taken into account. In this project, we have presented a depth study of all the parameters it have to be aware of, such as external mask design, change of disparity and vignetting. i.e. this thesis presents the details of all the steps we have to follow as far as the simplest thing such as the suitable sizes, shape and location for the optimal design of the external mask as far as the mathematical demonstration of the factors we will have to look after in order to achieve lightfield images.

Also, we used the built camera to capture pictures of some scenes and we computed a relative depth map of the scene. This was done using a photoconsistency model. We used the TV-L1 model for the optical flow [10][11]. The relative depth from the focused point is proportional to the movement caused by the perspective change between images. We computed the map depth as the optical flow. Although, we have achieved some good results, it is not worth for all the scenarios. The depth computation inherits all the optical flow computation problems, so, for obtain good depth map, the scene we are taking the photography must have some conditions. No uniformity is a must, in order words; all the objects that appear in the scene have to own textures and moreover have to have clear edges. Besides if we desire to achieve a depth map of a landscape, it would not be possible, because of the distance. Although the objects of the landscape have a distance between them, as they are really far from the focus point, the algorithm considers that proportionally they are on the same location.

However, for the scenes that have all the caprices of the optical flow desire, it have obtained good depths maps that show us properly which object of the scene was behind and which one was ahead.

For future work we should make a change in how we compute the disparity. We should find other techniques that does not limit us that much the scene, because it is not easy to accomplish all the requisites. One solution could be to compute the change of disparity through the structure tensor [12][13], this way what it is expected is not to be dependent on the conditions that optical flows forces to have, such us having textures and clear edges and simple scenes with not many objects.
REFERENCES


http://blog.lytro.com/post/132599659620/what-is-light-field

http://lightfield-forum.com/what-is-the-lightfield/

https://en.wikipedia.org/wiki/Light-field_camera
Annex

I. Pseudocode

Pseudocode of the function that creates the \( \chi \) mask for each \( I \):

\[
\text{for every pixel of } I \\
\text{if } I_{x,y}^{(i,j)} = 255 \text{ si es negre} \\
\quad \chi_{x,y}^{(i,j)} = I_{x,y}^{(i,j)} \\
\text{else} \\
\quad \chi_{x,y}^{(i,j)} = 0 \\
\text{end} \\
\text{end}
\]

Convert \( \chi_{x,y} \) to a binary
Pseudocode of the function that deletes all the vignetting in our images:

Convert $X_1$ to a binary
Convert $X_2$ to a binary

for every pixel of $I_{xy}$

if $X_1^{(i,j)} = 0$ and $X_2^{(i,j)} = 0$

$I_{xy}^{(i,j)} = \text{random value } 'a'$
$I_{x'y'}^{(i,j)} = \text{random value } 'a'$

elseif $X_1^{(i,j)} = 0$

$I_{xy}^{(i,j)} = I_{x'y'}^{(i,j)}$

elseif $X_2^{(i,j)} = 0$

$I_{x'y'}^{(i,j)} = I_{xy}^{(i,j)}$

end

end
Pseudocode of the function that creates the depth map:

for every $I_{x,y}$
Compute de module
Convolve it with a Gaussian

for every pixel of $I_{x,y}$

if $\lambda_x(i,j) \neq 0$
    $Auxiliar\ Matrix(i,j) = I_{x,y}(i,j)$
else
    $Auxiliar\ Matrix(i,j) = 77777777$;
End

if $Auxiliar\ Matrix(i,j) \neq 77777777$;
    $Acounterr(i,j) = Acounterr(i,j)+1$;
    $Sum(i,j) = Sum(i,j) + Auxiliar\ Matrix(i,j)$;
end

end

Depth map = Sum./Acounterr;
II. Lightfield images

Figure 89 Lightfield images of the scene illustrated on Figure 67