On how Hadamard matrices affect image quality and acquisition time in single pixel cameras

Author: Sergio González Casal.
Advisor: Artur Carnicer González.
Facultat de Física, Universitat de Barcelona, Diagonal 645, 08028 Barcelona, Spain.

Abstract: This paper presents a brief overview of the structure, advantages and working of a single pixel camera. This is information is presented in order to understand and explore the limits of this particular type of camera. With this purpose in mind, a Python simulation is used to study the resolution and the acquisition time.

I. INTRODUCTION

The most common cameras work with an array of pixels (usually a CCD) where every pixel receive the light corresponding to one point of the image. This is a problem when we try to get an image on the infrared, since pixels which detect this kind of light are more expensive than those that can sense visible light. For this reason, usage of a single pixel camera seems the best option in order to get cheaper infrared cameras.

The idea of a camera with only one sensor it is simply to take an spatial problem and convert it into a temporal problem. As it is explained in this paper, our camera biggest problem in front of the usual cameras is the time it takes to obtain an image of a good resolution. This problem appears from the fact that a single pixel camera needs to take a great number of measurements before creating an image, for reasons that will be explained later in this paper. A variety of methods have been studied [1] [2] to reduce this problem and improve this type of camera. However, in this case, we focused on the study of a very optimistic simulation with the purpose of making a simple approach to a subject still in development. Optimistic due to we do not take into account the noise or the SNR that would be a problem out of the simulation.

Until nowadays, the technology for constructing a useful single pixel camera did not exist. The physical limitation was huge, even if the mathematical base is relatively simple as it is shown in this paper. Nevertheless, with the arrival of Micro Electro Mechanical Systems (MEMS) and the improvement on computers, it is currently possible to obtain a good image in a reasonable amount of time. It is expected that with the development of all this technology mentioned above, the single pixel cameras will become more and more viable.

Summarizing, the single pixel camera has a great potential for improvement now and in the future. The motivation for this work can be resume in this potential and the fact that a pixel will be always cheaper than an array of them. A work based on Python and with an approach that is still useful to understand and study the basics on how and why a single pixel camera works, despite of its simplicity. Questions that are going to be answered in the next sections.

II. STRUCTURE AND BASIS OF A SINGLE PIXEL CAMERA

The goal of section A is to explain how to obtain an image using one pixel, while in section B the Hadamard matrices are introduced. Finally, in section C two structures for a single pixel camera are described.

A. OBTAINING AN IMAGE

With the purpose of explaining how to obtain an image, it is first necessary to talk about the concept of a set of basis vector. A set of basis vector is a group of vectors formed by a number of vectors equal to the dimension of the space, which multiplied by a coefficient and sum up can form any vector possible of this space.

To make it easier, imagine an image of 2x2, we can think of it like a vector of dimension four. If we have a set of basis vector, we can form any image of 2x2 combining them in the adequate way. This is exactly how a single pixel camera works, we put a mask, which is one matrix/vector, over our object letting pass the light only through some points. Just like this:

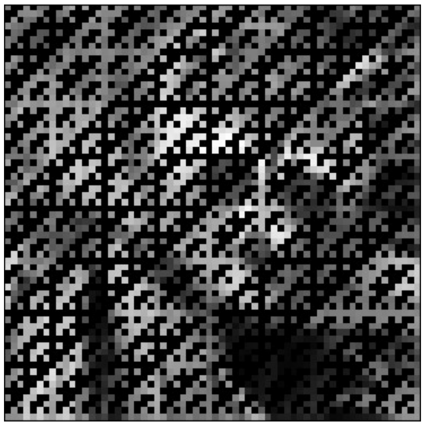


FIG. 1: An image covered with a mask.

Then, we measure the intensity of light passing through the mask and we repeat this process with the next matrix/vector of the basis until we have done it with all of them. If we have measured right, and following the theory, our image will be:

$$\mathrm{Im} = \sum_{i=1}^{NxN} I_i M_i \qquad (1)$$

Where Im is the image, M_i is the matrix of the basis number i and I_i is the intensity measured when we put M_i as a mask covering the image. We can think of the intensity as the weight that has one matrix in the final image.

All this have an important implication, there is no limit to the resolution of a single pixel. We decide it when we choose our set of basis vectors, the resolution of the image will be the dimensions of the vectors. However, it is important to note that more resolution implies more matrices and this

* Electronic address: sergio_g_c_1995@hotmail.com

implies more time to take an image, our problem is now a temporal problem.

In a single pixel camera our basis will be the Hadamard matrices, which are the issue of the next section.

B. HADAMARD MATRICES

The Hadamard matrices are used as a basis for generating the image in a single pixel camera, due to their special properties that makes them suitable for this purpose. Nevertheless, in this section it is only explained what they are and how to generate them, since the properties and applications of this matrices are too vast and out of the scope of this work. However, it is easy to find information about them as it is shown in the bibliography [3] [4].

A Hadamard matrix is a square matrix form by 1 and -1 as elements, in which every row has a half of the elements equal to the elements of the adjacent rows and the other half different. An example of one of the most basic Hadamard matrices will be the next one:

$$H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad (2)$$

In an image the -1 will be a covered region of the image and 1 a region of the image that let light pass. In other words, a Hadamard matrix used as a mask will be something like this:

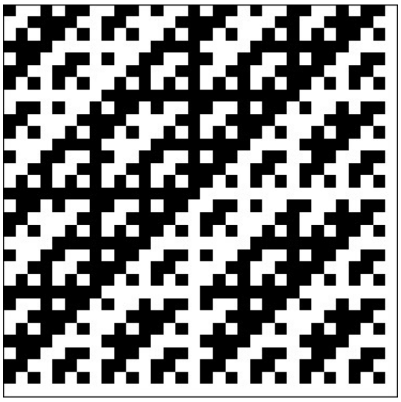


FIG. 2: A Hadamard matrix with dimensions of 32x32, shown as a mask.

The question now is how the simulation can generate any Hadamard matrix in a simple way. For fortune, there are recursive methods that allow the creation of a Hadamard matrix. In our case we use the Sylvester's construction, based on the next property. If H_N is a Hadamard matrix of dimensions NxN, we can generate a Hadamard matrix with dimensions 2Nx2N like this:

$$H_{2N} = \begin{pmatrix} H_N & H_N \\ H_N & -H_N \end{pmatrix} \qquad (3)$$

Where we can change where is the minus sign in order to obtain different Hadamard matrices. The equation (3) and a simple combinatory calculation to change the minus sign allow us to obtain a basis of Hadamard matrices. This is how the simulation does it, calculates the different combinations with the minus sign and generates the matrices with them.

It is important to make clear that there are other ways to define and construct a Hadamard matrix. However, in this work we take the easiest one due to this issue is a extensive one, as it is explained at the beginning of this section. Even so, with this approach to the Hadamard matrix we have enough for our single pixel camera to work.

Once the mask is generated there is only one problem left, how to cover a real object with the mask. In the simulation is easy because we are using digital images, but with a physical object is more complicated. In the next section two methods to cover a physical object with a mask are explained.

C. STRUCTURE OF A SINGLE PIXEL CAMERA

In this section two structures for a single pixel camera are shown. We will talk about one of the most basic and the one most used in our bibliography [1] [2] . Basically, because the first one is the most simple and similar to our simulation and the second one is the most used.

The first structure is based on a projector. Basically, we need the object to be in the darkness, once we have done this we project the Hadamard matrices over the object. In this case, the 1 in the matrix are illuminated while the -1 are kept in the dark. Finally, the intensity of light is registered with a sensor of light like a photodiode and we process this information as explained in subsection A.

This structure is simple but useful to make test and basic experiments due to the relatively cheap materials that usually are projectors repurposed [5]. This kind of experiments could be done using our simulation to generate the matrices and process the information, which could be an easy way to continue this work in the future.

In the second structure, the great difference is where we put the mask. While in the first structure we project the mask over the object, in this case we illuminate all the object and then, with a systems of lens, we focus the light into a DMD (Digital Micromirror Device). A DMD is a MOEMS(Micro Opto Electro Mechanical System) formed by an array of micromirrors that can vary its inclination between 10º and 12º using only electrical current.

Once the light is focus in the DMD, we can control where it goes changing the inclination in the micromirrors. Where there is a 1 in the Hadamard matrix, the mirror will direct the light to a light sensor and where there is a -1, the light will be directed to another point, for example to a heat sink. Finally, the image will be form like it was explained in subsection A. With the DMD we can also form a pixel with more than one micromirror, allowing us to make some studies like those presented in [2].

This structure is more expensive since you need a DMD for this purpose, but it is more rapid because a DMD responds faster than a projector. Even if they are very similar, the difference in speed is very important as it can helps with the acquisition time that is going to be discussed in the next section.

III. RESULTS AND DISCUSSION

In section A it is discussed if it is really necessary to use all the Hadamard matrices of the basis, while in section B the acquisition time of the camera is analyzed taking into account the number of pixel of the image and the quality for a resolution of 128x128.

A. NUMBER OF MEASUREMENTS

As has been explained above, to get a perfect image it is necessary to use all the Hadamard matrices of the basis. In this section we will focus on images of 128x128, which means a basis of 16,384 matrices. However, maybe using less it is possible to obtain and acceptable image, this will be the issue of this section.

To compare images the simulation uses the SSIM(Structural Similarity Index). This index allows the simulation to give an objective measure between the original image and the image we get, the maximum value of SSIM is 1 and the minimum -1. With it, we can now study how many measurements are needed to get an image. The original image is the next one:

FIG. 3: The original image of 128x128 which is the same that it is obtained using all the matrices of the basis.

First of all, notice that the simulation must use the Hadamard matrices without any concrete order, with the purpose of improving the quality of the images. This is mentioned because in the firsts simulations we obtained something like this:

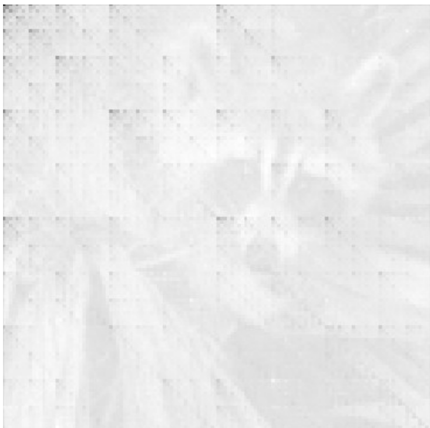


FIG. 4: An image of 128x128 pixels obtained using 14,000 of 16,384 possible matrices with a given order.

As you can see, there seems to be some squares and lines result of using the Hadamard matrices with order, which give some privileged directions. In the simulation, the order was the result of creating the combinations of the minus sign with recursive methods, but this was easily solve shuffling the combinations in order to make them random. If we use the Hadamard matrices without any given order, the result is the next one:

FIG. 5: An image of 128x128 pixels obtained using 14,000 of 16,384 possible matrices chosen randomly.

In this one we get some random noise, but is more similar to the original image as the SSIM clearly shows in the next plot:

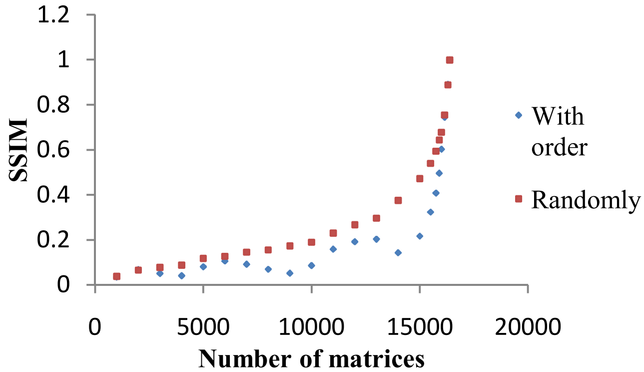
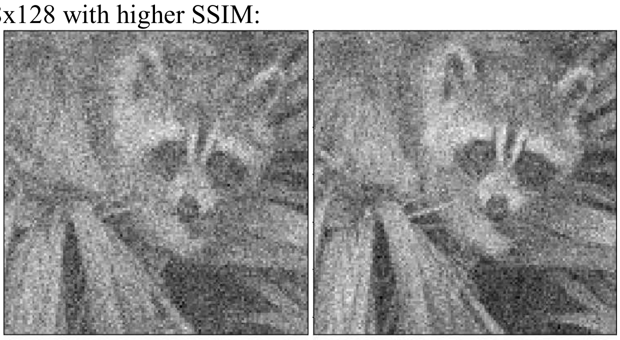


FIG. 6: Plot of the SSIM depending on the number of matrices for a 128x128 image, there are two representations: with order and randomly.

It is easy to notice that randomly is better and has a better tendency, since with order there are some strange variations in the SSIM. If we take as an acceptable value a SSIM of approximately 0.4(as the one above), then it is only necessary to use the 85% of matrices, which can make us win some time.

Besides, it is possible to slightly improve the image with random noise using filters. However, there are different types of filters with different uses depending on the situation, because of this, our work is not going to dedicate more time to this subject, only mention it as an option.

In order to help you understand what the SSIM really means in the above plot, we present some of the images of 128x128 with higher SSIM:



SSIM: 0.45 15,000/16,384 Matrices SSIM: 0.54 15,500/16,384 Matrices

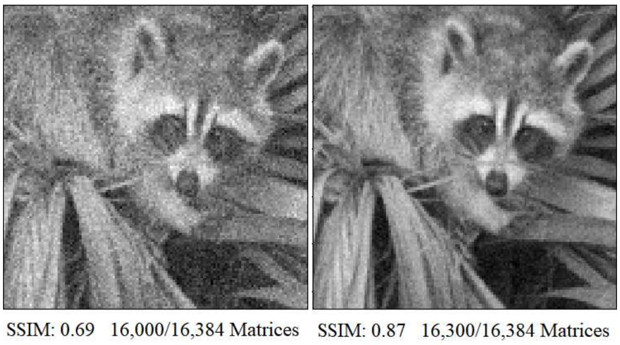


FIG. 7: Images of 128x128 pixels with their SSIM and number of matrices used. It is clearly seen how the quality of the image improves.

In a physical single pixel camera there will be also noise from the sensor and the usual defects of the devices. Therefore, in reality it is possible that 85% of matrices will not be enough, but that does not mean that all the matrices of the basis are needed.

B. ACQUISITION TIME

Before starting to analyze the data from the simulation, let us remind that the results presented in this section have been done with a personal computer in a simulation. In this particular case we use one nucleus from a processor Intel Core i5-4690 with 3.50 GHz and with 8 GB of RAM(Random Access Memory).

This fact implies two things, the first one is that different processors would get different time, while the second one is that a real single pixel would take more time because it needs to generate a real mask over the object. Therefore, our results are useful to make an study of a single pixel camera, but would change with those factor mentioned above.

When studying the acquisition time with the simulation, two studies are developed. One with all the matrices and another with the 85% of matrices, due to the results obtained in the previous section. The resulting plot is the next:

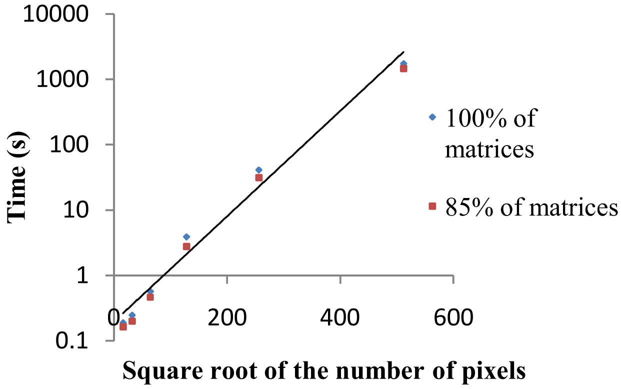


FIG. 8: Plot of the acquisition time as a function of thesquare root of the number of pixels in the image. There are two representations, one with all the basis and another with the 85% of it. In addition there is an exponential trend fitted for the 100% of matrices.

As is clearly seen there is an exponential relation between time and the number of pixels. It is also clear that the variation of time using the 85% of the matrices is as little as expected. If we make the exponential representation to the 100% of matrices the result is the next function:

$$t = 0.196 * e^{0.018*\sqrt{\#pixels}} \qquad (4)$$

This function simply verifies what has already been stated in this paper, the time is the real problem for the single pixel camera. With a dependency in the number of pixels, any attempt to get a better resolution increase acquisition time exponentially. This can clearly be extrapolated to a real single pixel camera, because its time growth would be equally exponential. Due to the source of the problem, which is the need for project more matrices in order to get better resolutions. All this implies that, currently, we can only work with single pixel cameras of low resolution.

Finally, we studied the relation between SSIM and time, to see the quality related to the acquisition time. In order to carry out this study, we focus on an image of 128x128 because its acquisition times are not too big or too little. The plot obtained is the next one:

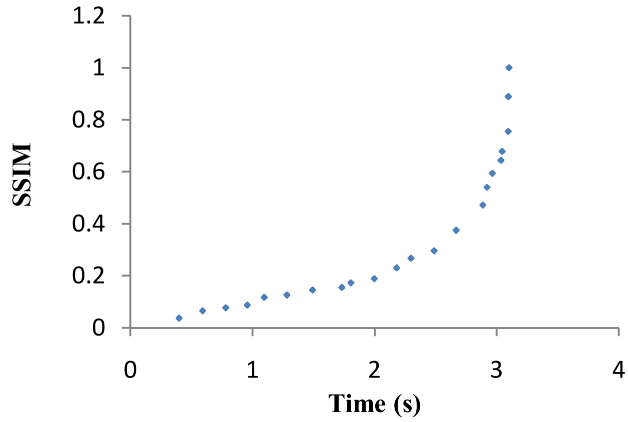


FIG. 9: Plot of the SSIM as a function of time for an image of 128x128 pixels.

The result is very similar to the one exposed in the earlier section, due to the relation between time and the number of matrix. However, new information can be obtained from this plot. As it can be seen, it takes about 0.5 second less to get the image if a 0.4-0.3 is taken as a valid value for SSIM. It is not a big time, but it is an improvement.

IV. CONCLUSIONS

After creating the simulation of the single pixel camera and studying its results we arrived to the next conclusions:

- The single pixel cameras are more than a useful option to get an image on the infrared, and with the current technology there is plenty of space to develop them.
- The main limitation of a single pixel camera is the time required to obtain an image. However, images of low resolution are nowadays achievable in a reasonable amount of time.
- In order to obtain a perfect image, we need as many measurements as pixels has the image. But we can obtain a relatively good image with less measurements, reducing the time of acquisition.

- A single pixel camera could be created only with the simulation used in this work and a projector. This could be an interesting way to continue this work, since generating the Hadamard matrices is the first step to create a single pixel camera.

V. ACKNOWLEDGEMENTS

I would like to thank my advisor Artur Carnicer for guiding and supporting me through this work and my family and colleagues, because explaining them how a single pixel camera works helped me understand it better. Finally, I would like to give thanks to the online community of stack overflow which offered me some advice in creating the simulation.

---

[1] Mark A. Davenport, Dharmpal Takhar, Jason N. Laska, Ting Sun, Kevin F. Kelly, and Richard G. Baraniuk Marco F. Duarte, "Single-Pixel Imaging via Compressive Sampling," *IEEE SIGNAL PROCESSING MAGAZINE*, vol. 25, no. 2, pp. 83-91, March 2008.

[2] Matthew P. Edgar, and David B. Phillips Ming-Jie Sun, "Improving the signal-to-noise ratio of single-pixel imaging using digital microscanning," *OPTICS EXPRESS*, vol. 24, no. 10, pp. 10476-10485 , May 2016.

[3] K. J. Horadam, *Hadamard Matrices and Their Applications*, 1st ed. New Jersey, United States of America: Princeton University Press, 2006.

[4] Eric W. Weisstein. MathWorld. [Online]. http://mathworld.wolfram.com/HadamardMatrix.html

[5] Wictor Lund, and Jerker Björkqvist Stefan Grönroos, "A Single Pixel Camera based on a DLP Video Projector," *Proceedings of Automaatio*, vol. XXI, no. 44, pp. 1-6, 2015.