Computational Search of the Lowest Bound of Phase Information for Plausible Digital Image Reconstruction

Mateo Laguna Guantiva

Advisor: Ph.D. Jaime E. Forero
Ph.D. Alejandra C. Valencia

Physics Department
Universidad de los Andes

This dissertation is submitted for the degree of
Physicist

Universidad de los Andes 2018

I would like to dedicate this thesis to ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

Mateo Laguna Guantiva
2018

Acknowledgements

And I would like to acknowledge ...

Abstract

Digital images are a fundamental topic in the information era due to its relevance in multiple disciplines. This monograph investigates one fundamental aspect of a digital image: the phase information. In this document, I show how the phase information is relevant for digital image reconstruction. This computational experiment is made under the support of artificial neural networks as the judge who evaluates if the reconstruction of an image is plausible or not. I find the minimal amount of phase information for different sets of images where the neural network fails the recognition process. This minimal amount of phase information is very low compared to the rest of information available in the image so phase information is relevant for a digital reconstruction process.

Table of contents

List of figures

Nomenclature

Roman Symbols

ANN	Artificial Neuronal Network
DFT	Discrete Fourier Transform
DIR	Digital Image Reconstruction
IDFT	Inverse Discrete Fourier Transform

Chapter 1

Introduction

1.1 State of The Art

Images are one of the main ways that human beings are able to communicate their ideas, feelings, and thoughts since many centuries ago. Just as every single aspect in daily life was affected due to the internet and information era, so images also were affected, creating a new concept called digital images. As technology grew, digital images became a fundamental pillar for multiple branches such as communications [1], science [2], industry [3], military technology [4], politics [5], artificial intelligence and medicine [6], etc. Hence, the study of digital images became a serious matter not just to scientists but also to other many disciplines.

There are two main processes that motivate the study of digital images: *digital image reconstruction* and *digital image compression*. Both processes deal in some way with the loss of information and behind them there is a fundamental question that must be answered to carry out the processes: **where is the crucial information in the digital image?**

If there is information in the image that is not relevant or does not contribute widely to the image visualization process, then that information could be removed and the size of the information that must be transmitted will decrease just as the computational resources. On the other hand, if there is crucial information in images then if an image is received not having this information it could be predicted that it is not possible to achieve a digital image reconstruction process. This process is called in the academic world as Compressive Sampling which is defined by [7] as *if it was possible to directly acquire just the important information about the object of interest.*

This fundamental question was explored almost 40 years ago by Monson H. Hayes, Jae S. Lim and Alan V. Oppenheim in their *Signal Reconstruction from Phase or Mangitude* article [8] where they explore how to reconstruct a signal based on phase and magnitude information in the spectral domain. This article is relevant because a powerful mathematical tool as the DFT was introduced to help answer the fundamental question. One year later, Jae S. Lim and Alan V. Oppenheim publish an article entitled *The Importance of Phase in Signals* [9] where the idea that the phase has the relevant information in a signal was introduced for the first time.

After those articles were released, phase information in an image got the attention of scientists. Phase was begun to be analyzed in speech or images as in *Phase in Speech and Pictures* [10] or in optical imaging processes as in *Phase Retrieval with Application to Optical Imaging* [11].

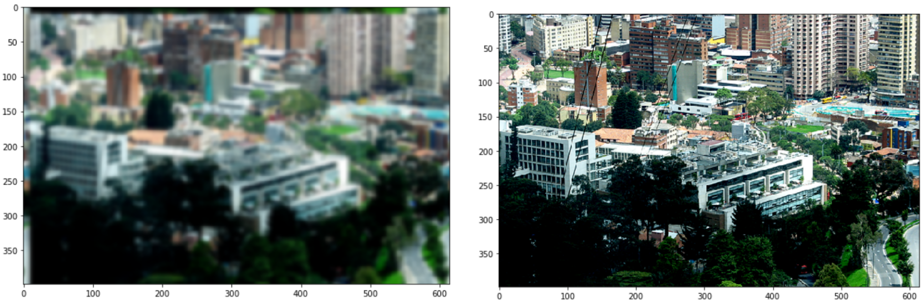
Thus, it was clear that phase was relevant in a digital image. Hence, it was time to let this knowledge about the relevant role of phase information and spectral domain in digital images will enhance the digital image reconstruction process as in *Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information* [12] and *Dgital Image Reconstruction in the Spectral Domain Utilizing the Moore-Penrose Inverse* [13].

The main idea for this dissertation was established: phase information of an image is relevant and is able to help to enhance the digital image reconstruction process. Nevertheless, in numbers, how much more relevant is phase information than frequency information (i.e. phase and amplitude information)? and more crucial, what is the lowest bound of phase information that is needed to be able to reconstruct an image?

1.2 Digital Image Reconstruction

There are two ways to understand what is Digital Image Reconstruction. The first way comes from one of the main books of the field of digital images: *Digital Images Processing* written by Gonzalez and Woods. In the book DIR process is defined as *to reconstruct or recover an image that has been degraded by using a prior knowledge of the degradation phenomenon.*

As the reader can see, the definition is given above of DIR fits so well to the previous example: there is a recovery of an image that has been degraded by using the appropriate filter (another example of digital image reconstruction process as the use of restoration techniques see [14]). However, even if the information in Figure 1.1 (a) (taken from [15] with prior authorization) is *dirty* (i.e. it has some noise), the information is still there. Thus, even if the definition fits well for the field of digital image processing, it is an incomplete definition because the definition is implicitly saying that the information is still on the image but degraded (i.e. lost of quality), but what if the information is missing?



(a) A smoothed image (b) Original image

Fig. 1.1 Example of a digital image reconstruction process

Reconstructing an image attacking the degradation process that was made on it is a procedure that is assuming that information is not lost but dirty. Hence, trying to reconstruct an image that has lost information (not just noise on the information) without any previous knowledge of how was the information is a more general case for digital image reconstruction process.

This more general case of digital image reconstruction is presented in the current document. The main idea is to reconstruct an image after a phase information removing process without any restoration technique. The concern of this dissertation is to find the lowest amount of phase information that is needed to reconstruct an image, so there is no intention to create a technique to know how was the original information but to make a *plausibile* (this word will be explained in section 1.3) reconstruction of an image without the phase information that was removed.

Consider a landscape puzzle and the main concern of knowing the theme of the puzzle. Some pieces of the puzzle are removed, the reconstruction process is trying to join the remaining pieces and with them make the puzzle in such a way that it can be perceived that it is a landscape (i.e. giving sense to the remaining pieces, it means, there are some pieces that are crucial to know that is a landscape, as an example the sun piece). In the specific case without loss of information, all the pieces of the puzzle are present but some of them have been scratched with a marker, so the reconstruction process is to clean and join the pieces to know that is a landscape. Note that there is no need to give sense to the pieces because they are complete and joining them will be enough to see that is a landscape.

1.3 Artificial Neural Network: TensorFlow

The title of this dissertation is *Computational Search of the Lowest Bound of Phase Information for Plausible Digital Image Reconstruction*. The term **Digital Image Reconstruction** was explained in the previous section, now it turns to explain what **Plausible** word means.

When this document was planned the main idea was to explore where the crucial information of an image was stored and this will be done removing information until the image was not recognizable. But there was a problem in that thought: who will decide if the image was not recognizable anymore? Letting this task to a human being could fall in a matter of subjective, some people could consider that the image is still recognizable, some other do not. This is where the artificial neuronal network come into play. Thus, an impartial judge was needed. The neuronal network [1] will have the burden of deciding if the image is recognizable or not, in other words, the neuronal network will have the responsibility of defining the word *plausible* for digital image reconstruction process.

Therefore a new question is born: what is an artificial neuronal network? According to [16] *Artificial Neural Networks are relatively crude electronic models based on the neural structure of the brain*. In other words, a neuronal network is just a new technology in the computer field to solve problems; its name comes from the similarity that shares with the human brain: a neuronal network is made of neurons just an artificial one and the artificial neuronal network learn from patterns just as a biological neuronal network learns from experience. Perhaps behind the success of neuronal networks is that they are overcoming the edge of the binary computation (e.g. just as the quantum computer that is crossing the binary barrier). According to [16]: *Together these neurons and their connections form a process which is not binary, not stable, and not synchronous.*

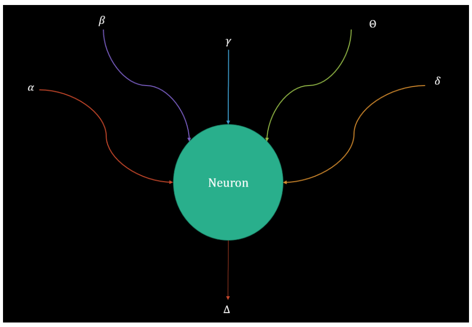


Fig. 1.2 Basic scheme of a neuron of an artificial neuronal network

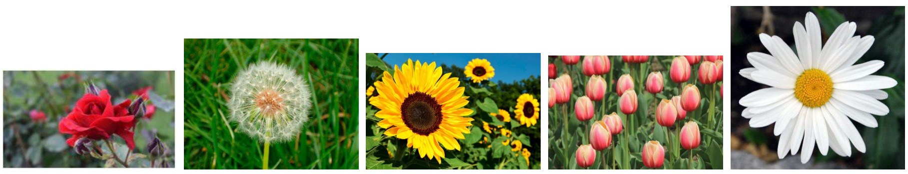
[1] from now on the term *neuronal network* makes reference to an artificial neuronal network controlled by a computer and not a biological one

The fundamental pillar of a neuronal network is a neuron. From the medical field, it is known that a neuron receives inputs via electric impulses do something inside with this information and send an output. A neuron of an artificial neuronal network works as in shown in Figure 1.2, each color represents the weight of the input (greek letters in each arrow: α, β, γ, δ, θ), not all the inputs have the same importance. Inside the neuron occur two processes: sum and transfer. The summing process decides how to compute the inputs that depend upon the use that will have the neuronal network (i.e. neuronal networks for speech recognition are different from robot learning neuronal networks). The transfer process is the way to decide how the input will be sent (e.g. discrete or continuous values), in other words, the result of sum process is sent to a real output that is useful for humans.

Nevertheless, neuronal networks are not made by one single neuron but a bunch of them. Here comes the concept of layers, the neuronal network is structured in layers, each layer has many neurons that are connected to other neurons from other layers with specific sum and transfer functions. It is important to recall that a bunch of neurons in layers without well-defined communications channels will not work as a neuronal network.

The last part of a neuronal network is the training. There are two types of training: supervised and unsupervised. In the first one, the neuronal network is provided by the correct answer and based on that the performance can be graded. In the second one, the neuronal network has to the task by itself without any help from outside.

In the present computational dissertation, the creation of a complete new neuronal network was a burden that was left to Google. Google provided free access to the code of an artificial neuronal network called Tensorflow that is used to image recognition. In just less than half an hour the neuronal network can be trained (supervised training) and be ready to start the classification process. In the simplest case of Tensorflow, the network is trained to recognize five entities of images: rose, dandelion, sunflower, daisy, and tulip.



(a) Rose (b) Dandelion (c) Sunflower (d) Tulip (e) Daisy

Fig. 1.3 Set of entities in the world of the Tensorflow neuronal network (Figures from Tensorflow under the creative commons license)

After the training time of the network, it is ready to work. How does it work? Through a command line in python, an image is sent to the Tensorflow neuronal network (suppose the image is a sunflower just as Figure 1.3 (c)), what should be expected? (to know in detail the explanation about how this neuronal network of Tensorflow works, see [17])

```
sunflowers (score = 0.98011)
daisy (score = 0.01123)
dandelion (score = 0.00593)
tulips (score = 0.00210)
roses (score = 0.00063)
```

Fig. 1.4 Classification result made by the neuronal network upon receiving a sunflower

Finally, the necessity of using an artificial neuronal network is because a human can bring uncertainty to the computational experiment due to the subjectivity behind the term *plausible* when is time to define how was a digital image reconstruction process. Besides, human being vision as a classification system was overcome by InceptionV3, the engine behind the neuronal network of Tensorflow. Thus, trust classification task to the artificial neuronal network allows guaranteeing objectivity in the classification of images.

1.4 Importance and Applications

A topic becomes relevant when it appears in many fields. Consider mathematics, its huge relevance is due to the fact that is almost in any field: physics, astronomy, biology, industry, economy, computing science, even art. Thus, the study of maths is not just a vocational matter but a necessity. It is the same case as digital images, the information age where humanity is currently involved transformed the concept of an image and this new concept became essential to human daily life due to the many fields where digital images are present. In this section, those branches that were just cited in the state of art will be explained in more detail.

In **communication** field the digital image processing is a crucial task for many powerful companies such as Instagram, WhatsApp, Facebook, Google, Microsoft, IBM, etc due to the huge amount of images that these companies must handle every day to keep working. A company of communication service just as WhatsApp should be focusing on the quality of the communication process and this includes image messages that are highly sent in this application. Thus, studying the information behind an image is not just an academical matter but a business requirement.

In **science** branch digital images are fundamental for many disciplines. In astronomy, many observatories such as the Paranal Observatory do investigation with huge telescopes but the results that those big artifacts get are images from the sky or space if they come from satellites. In medicine images become an issue of life or death: seeing in a tomography a tumor at the right time could mean saving the life of the patient. Thus, exploring how to enhance the quality of images or how to recover information that was lost in the communication process is crucial because it means scientific knowledge.

In **industry** field digital images represent money. As an example, consider the case of an artist that creates a digital comic, his or her work represents time, knowledge and effort and that is the reason why they charge people to see their creations. Now consider the case where the digital comic begins to be exploited by someone extern of the comic creation and also this subject receives money for it. Thus, copyright protection in digital images becomes a crucial topic and many disciplines as cryptography, signal compression and digital images processing [3] have to work together to avoid industry lost of money.

In **military technology,** branch it is just necessary to read [4] to see how the US army has been exploiting the research in digital images processing field to enhance and produce new military technologies. Some examples that reader will find in [4] of these technologies based on digital image processing are *remote sensing*, *computational vision* and *autonomus vehicle*.

The last military technology also belongs to **artificial intelligence** field where digital image processing has become fundamental. Recognizing images in real time by cameras is the basic mechanism to the autonomous vehicle technology strongly researched by Google. Another example of the use of digital images processing is in artificial intelligence applied to robotics (the artificial neuronal networks that the robots use to work have extensive layers of neurons that deal with image digital processing to understand and respond to the environment). Companies as IBM and Facebook are investing an extensive amount of capital to the research of this technology.

The last field, **politics**, could be considered as a completely disconnected branch of the digital image processing. However, [5] claims that images have power in this digital age, it means that digital images are not just a tool of communication, a tool of learning, a tool of productivity or a tool of technology but a tool of political power. Thus, politics field that seems far away from maths, computing and information theory is actually taking advantage of this digital image processing branch.

Hence, this dissertation document is made having understood the main concern of the industry and science fields to achieve a deeper knowledge of digital images branch. Studying where is stored the crucial information in an image could revolutionize many areas in the current world. In this way, the dissertation thesis of this document is not just a great research academic objective but could bring new lights to many branches in the industry.

Chapter 2

Theoretical Framework

2.1 What is a Digital Image?

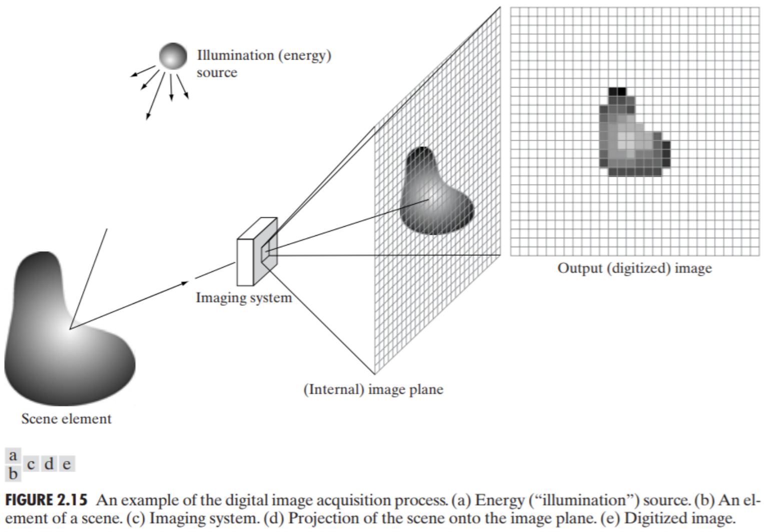


Fig. 2.1 Digital image acqusition process

Before defining what a digital image is, it is important to know how a digital image is created. Figure 2.1 (a) (taken from [18] Chapter 2, section 1, page 50.) shows the basic and initial component to create a digital image: the source. There must be a source of illumination that is radiating energy, in the day it would be the sun as an example. This source of illumination will radiate energy to many objects including the one that is in the scene, as Figure 2.1 (b) shows, that will be imaged. The scene element will reflect some of the light that comes from the source and will impact the imaging system as in Fig 2.1 (c). Once the light reaches the sensor (imaging system) there is a conversion process. The input process is Figure 2.1 (d) where the input light coming should be transformed to scalar values as an internal process (i.e. energy reaching the sensor should be transformed to positive scalar number). Then Figure 2.1 (d) is the output process where the image will be stored in digital format but scalar values of the internal process made by the sensor should be transformed to 0 and 1 values due to binary system with which computers work.

Following the definition provided by [18] a digital image could be seen as a *two-dimensional function of the form $f(x,y)$ where the value of f in coordinates (x,y) gives a positive scalar quantity that is proportional to energy radiated by a physical source (e.g. electromagnetic waves)*. In other words, a digital images is a two-dimensional matrix (for the reason explained in the previous paragraph a 3D object is imaged in 2D screens) where each (x,y) position there is a value proportional to intensity of light (as it was explained in the above paragraph this task is done by the sensor) that is a non-negative value.

Until now there was a brief description of the physics behind the image capture process. However, how can digital images be represented? After the acquisition of the image by the sensor there is a process called *sampling and quantization* (for further details see section 2.4.3 in [18]) that in few words transform input energy of the light values to discrete values and returns a 2D matrix. Each of the points in Figure 2.2 (Figure from [18] Chapter 2, section 4, page 55) is called *pixel* and is the basic unit of a digital image. Each pixel has a real and positive value that represents the intensity of light reflected from the object imaged. The following representation of a digital image is highly important due to that will be used along the current document (all the algorithms presented in Chapter 3 are based on this digital image representation):

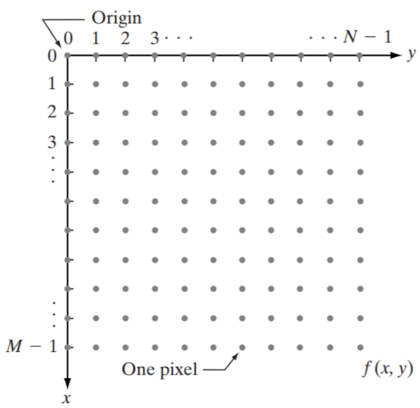


Fig. 2.2 Computational representation of a digital image

2.2 Fourier Mathematical Theory

Jean Baptiste Joseph Fourier made an important work that is currently applied in many disciplines almost two hundred years later after he published his revolutionary ideas. One of the main thoughts that Fourier had, quoting [18], is that *any function that periodically repeats itself can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient*, this is well known as Fourier Series. Furthermore, *it does not matter how complicated this function is; as long as it is periodic and meets some mild mathematical conditions, it can be represented by such a sum.*

This great conclusion was extended some years later to functions that are not periodic but are integrable this let Fourier transform tool arose. However, besides the huge aid of Fourier theory to break down a complicated function into sines and cosines, is there any other advantage? Indeed, it is. It turns out that the spectral or frequency domain is so helpful to make some useful operations but the most important point of Fourier theory is that there is not a cost function associated to pass from the time domain to the frequency domain or vice versa.

Before handling advantages of frequency domain it is important to recall the mathematic expressions of Fourier theory (where $i^2 = \sqrt{-1}$):

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi ux} dx$$

Eq. 2.3 Mathematical expression for 1D Fourier transform

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{i2\pi ux} du$$

Eq. 2.4 Mathematical expression for 1D inverse Fourier transform

However, signals that are relevant for this document are not in one dimension but in two. A digital image is, as Figure 2.2 shows, a matrix which means that to apply a Fourier transform it is necessary to extend the concept to two dimensions as is shown in Equation 2.5 and 2.6 (for further details about Fourier theory see [18] Chapter 4, section 2).

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-i2\pi(ux+vy)}dxdy$$

Eq. 2.5 Mathematical expression for 2D Fourier transform

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{i2\pi(ux+vy)}dudv$$

Eq. 2.6 Mathematical expression for 2D inverse Fourier transform

2.3 Two-Dimensional Discrete Fourier Transform

Fourier transform was explained in the previous subsection. However, signals such as an image are not limitless functions but finite. Thus, there is a necessity to define the Fourier transform in a finite case, that is called *Discrete Fourier Transform.* In the case of an image the Fourier transform should move along all the pixels that are finite (i.e. $f(x,y)$ where $x = 0,1,2...N-1$ and $y = 0,1,2...M-1$).

$$F(u,v) = \frac{1}{NM}\sum_{x=0}^{N-1}\sum_{y=0}^{M-1} f(x,y)e^{-i2\pi\left(\frac{ux}{N}+\frac{vy}{M}\right)}$$

Eq. 2.7 Mathematical expression for 2D discrete Fourier transform

1

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{M-1} F(u,v)e^{i2\pi\left(\frac{ux}{N}+\frac{vy}{M}\right)}$$

Eq. 2.8 Mathematical expression for 2D inverse discrete Fourier transform

The way that equation 2.7 must be used is: start with $u = 0, v = 0$ and replacing these values on the equation 2.7, sum over all values of x and y then the equation will throws the DFT for $u = 0, v = 0$ (i.e. $F(u = 0, v = 0)$) that will be a complex number.

Fourier theory has a lot of properties that are useful in many areas. However, there is a special property that will make the computational experiment less expensive in computational terms.

Symmetry is a property that is present in many systems and when there is symmetry some features on the system are satisfied. In this case, symmetry plays an important role because it will decrease the computational time and processing cost.

The symmetry that is present in this case is called *conjugate complex symmetry*. This symmetry is satisfied due to the condition of a *real* value input (as it was explained previously the input is a 2D matrix of real values because each value is proportional to the energy of the reflected light on the object). Conjugate symmetry means that ($*$ denotes the complex conjugate (i.e. the complex number with imaginary part with opposite sign), N the last index of frequencies (i.e. 48400 in the case of Fig 3.1), $|_j$ represents the complex number at index j and $f(x)$ represents the complex value of the 2D matrix after the DFT):

$$f(x)|_{N-n} = f^*(x)|_n \quad n = 1,2,3...$$

Eq. 2.9 Conjugate symmetry condition

The true value of the last equality can be proved by making use of definitions. Consider the value $f(x)_{|}n$ and $f(x)_{|}N-n$ as it was explained previously using the DFT:

$$f(x)|_n = \sum_{j=0}^{N-1} x_j e^{-i\left(\frac{2\pi jn}{N}\right)} \tag{2.1}$$

$$f(x)|_{N-n} = \sum_{j=0}^{N-1} x_j e^{-i\left(\frac{2\pi j(N-n)}{N}\right)} = \sum_{j=0}^{N-1} x_j e^{-i2\pi j} e^{\left(\frac{2\pi jn}{N}\right)} \tag{2.2}$$

Now, it is useful to remind that $e^{-i2\pi j} = cos(2\pi j) - i \times sin(2\pi j) = 1 \quad \forall j \in \mathbb{Z}$, then:

$$f(x)|_{N-n} = \sum_{j=0}^{N-1} x_j e^{i\left(\frac{2\pi jn}{N}\right)} = \sum_{j=0}^{N-1} \left(x_j e^{-i\left(\frac{2\pi jn}{N}\right)}\right)^* = f(x)^*|_n \tag{2.3}$$

The last expression means that after DFT is executed on Figure 3.1, if $f_1 = f(\nu = 1, \eta = 0) = 5+5i$ as an example, then by equation 2.9 $f_2 = f(\nu = 220, \eta = 220) = 5-5i$. Now, consider that f_1 and f_2 are the red squares:

Hence, the original image has N real values of information. In DFT domain it seems that the there is 2 times more information due to imaginary values (i.e. $2N$). However, as the reader can see, with the conjugate complex symmetry there is only N of real information

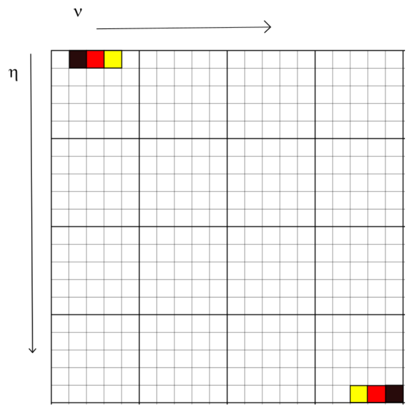


Fig. 2.10 Conjugate complex symmetry on the 2D matrix after DFT (squares of the same colour satisfy equation 2.9)

because the one half is the same as the other one but opposite sign in the imaginary part (i.e. there is no new information in the DFT domain).

At this moment the algorithm just has to work on half of the frequencies and apply the same process to the respective complex conjugate (i.e. algorithm does not have to walk over the whole matrix but just $\frac{N}{2}$ values). However, the article by [19] established that there is no need to walk on $\frac{N}{2}$ but just $\frac{N}{4}$ values. This new symmetry is not due to maths but computation. The DFT and IDFT are two transforms that computationally are expensive in time and processing resources. For that reason, there is FFT (*Fast Fourier Transform*) that executes the same transform but optimal. (more information about the algorithmic complexity of DFT and IDFT and computational symmetry see [19]). Hence, the algorithm just has to work with $\frac{N}{4}$ frequencies and do the same process in the respective complex conjugate and computational conjugate.

2.4 Phase and Amplitude Information

Right away, it is important to reconsider a basic property about complex numbers. A complex number is interpreted by the computer in cartesian coordinates (e.g. $a+bi$, where a is the **real** part and b the **imaginary** one). However, every complex number can be rewritten in polar form (i.e. $|A|e^{i\phi}$) where A is known as the magnitude of the complex number (see equation 2.11) and the ϕ is known as the phase (see equation 2.12) of the complex number.

There is a crucial question: why phase or amplitude is relevant in terms of information if it is just another way to represent the same time domain information? Indeed, there is not a new creating information process, but when information of the time domain is represented in

$$|A| = \sqrt{a^2 + b^2}$$

Eq. 2.11 Magnitude or spectrum of the Fourier transform

$$\phi = tan^{-1}\left(\frac{b}{a}\right)$$

Eq. 2.12 Phase angle or phase spectrum of the Fourier transform

the frequency domain through concepts like phase and amplitude there is one of those terms that is more relevant than the other one.

In Figure 2.13 (the figure was taken from [20]) DFT is applied to both pictures, then phase information of both pictures is swapped with each other, after applying IDFT to both pictures the previous computational experiment conclude how relevant is the phase to reconstruct the original image.

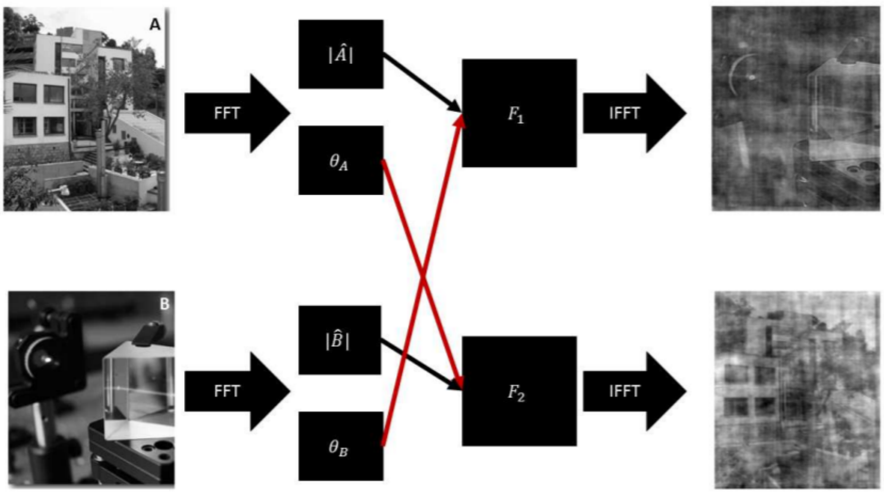


Fig. 2.13 Computational experiment to test the relevance of phase

In this way, the main idea of the dissertation is established: phase information is relevant to achieve a good digital image reconstruction process as can be seen in Figure 2.13. However, is there any way this relevance can be quantified? and even more, **what is the**

minimum amount of phase information needed to achieve a *plausible* [2] digital image reconstruction process?

[2] to see what this word means see Chapter 1, section 1.3

Chapter 3

Methodology

3.1 Prior Study of the Image Information

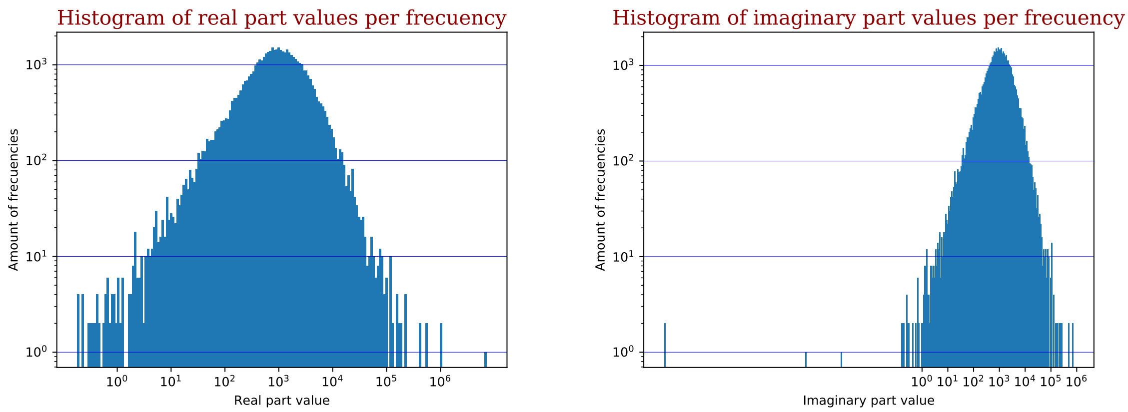
Before starting the study about if phase information is relevant or not to the digital image reconstruction process, it is mandatory to study what the image has in terms of information previous to the work that will be done on the image.

As the point of start, it will be used the Figure 3.1 to do all the analysis of information in images. At the end of the research, the analysis will be extended to a set of different images. (The use of a grayscale image is to achieve a reducement in the computational cost)

Fig. 3.1 Target image for the information research process

In chapter 2 it was shown that an Image is a set of entities called pixels and that every pixel has a real value associated that represents its intensity. Also, it was shown that a DFT can be applied to the image and it will produce an image with a set of *pixels* that now have complex values. Before starting handling the image, it is important to know how these *pixels* in the frequency domain are stored in the image: do they have a predictable behavior or a random one? the real and imaginary values of the *pixels* in the frequency domain have any upper or lower limits? how is the distribution of these values? and more crucial, do these values repeat along the image?

Taking Figure 3.1 and applying DFT gives a set of real and imaginary values. In this case, as the image has a dimension of 220x220 pixels that means that will produce a list of 220 real and imaginary values per each row (220 rows in total). Thus, the following displayed two images show the real and imaginary value of each frequency (i.e. 48400 frequencies in total).



(a) Histogram of real values of Figure 3.1

(b) Histogram of imaginary values of Figure 3.1

Fig. 3.2 Histograms of real and imaginary values of Figure 3.1

In the two previous plots, it can be seen that most of the real and imaginary values of frequencies lie on not extreme values (i.e. there are just a few frequencies with extremely high values or extremely low values). As the plots show, *pixels* values follow a distribution, it means there is a chance to model the probability distribution of the values of the frequencies (that would be a good objective of research but is beyond the scope of this dissertation).

Although the two previous histograms tell the reader in which rank of values lies every single real and imaginary value of each frequency, histograms do not tell how many frequencies lie in each rank. For instance, how many frequencies have real values greater than 10 but less than 100? how many frequencies have imaginary values greater than 100.000 but less than a million? how are the frequencies distributed in the different *sections* along the image?

According to the previous plot, the majority of frequencies of Figure 3.1 lie on two specific sections: almost half of the total frequencies have a real (and also imaginary)

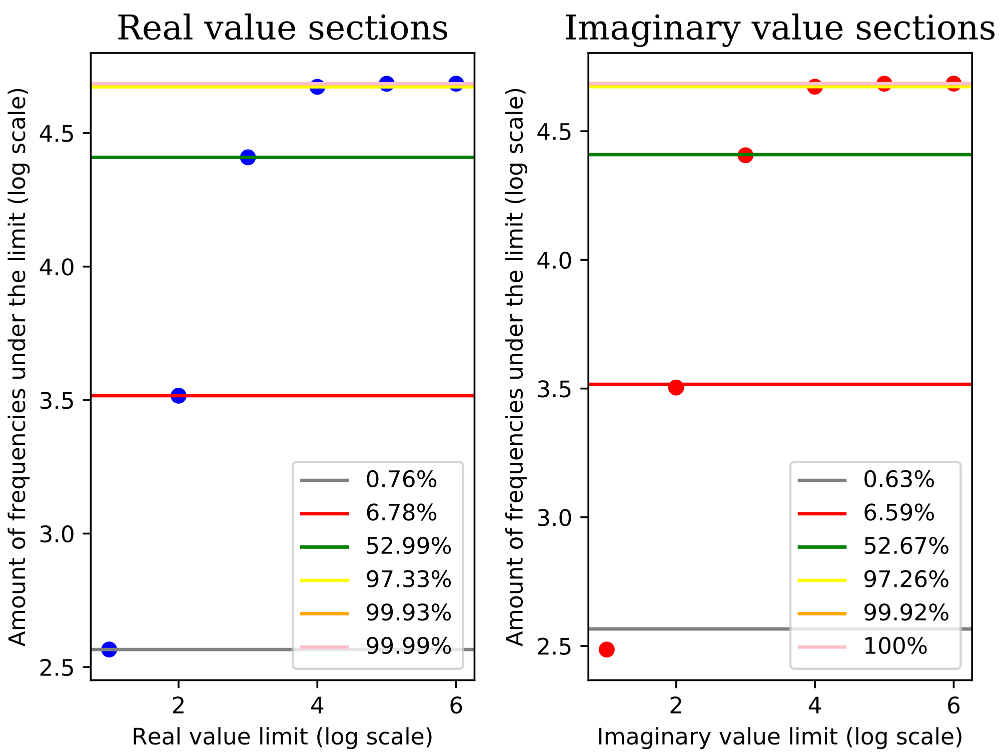
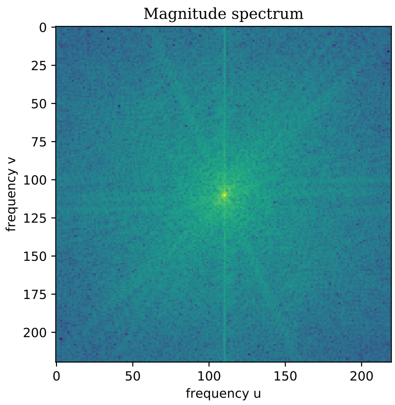


Fig. 3.3 Real and imaginary values in the DFT domain of Figure 3.1

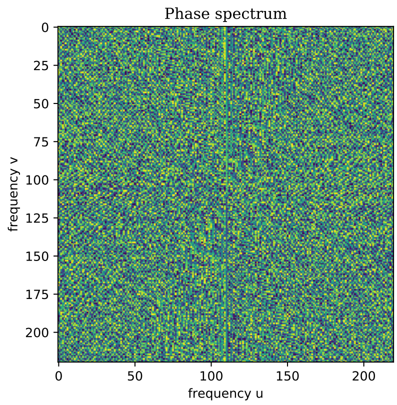
value less than 1000 and greater than 100, the other half of frequencies have a real (and imaginary) value greater than 10.000 and less than 100.000. Other zones have a low amount of frequencies.

There are two other important plots to visualize previously to the image manipulation: magnitude and phase spectrum. As their names indicate, both plots show those quantities. However, these 2D plots are not completely understandable for a person, hence a 3D plot provides more information about the nature of the original image.

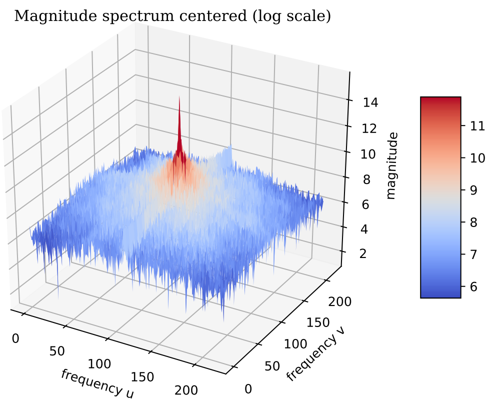
In Fig 3.4 (a) there are few changes in its visualization. The zero frequency (also known as the mean of the magnitude values) is not at the left top corner (as it is supposed to be) but in the center of the image (this is useful because it let the low frequencies at the center using the symmetry properties mentioned in chapter 2, so it is easier to interpret the image). Also, the magnitude spectrum is shown on a logarithmic scale because of its huge range of values. (magnitude and phase information is defined in Chapter 2, section 2.4)



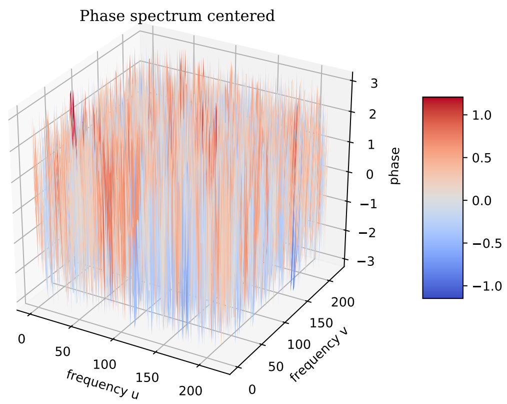
(a) A 2D plot of magnitude values of Figure 3.1



(b) A 2D plot of phase values of Figure 3.1



(c) A 3D plot of magnitude values of Figure 3.1



(d) A 3D plot of phase values of Figure 3.1

Fig. 3.4 Magnitude and phase spectrum of Figure 3.1

3.2 Algorithm Construction

3.2.1 Isolated Frequencies

Until now it is known the real and imaginary values of frequencies in the Fourier domain, where are they concentrated, how they fit a distribution, etc. Nevertheless, it is time to know whether phase information in the frequency domain is crucial in an image reconstruction process or not. The first approach is to analyze what is the importance of **every single** frequency in the DFT domain.

This subsection is called *isolated frequencies* because the algorithm that is executed finds the relevance of each frequency in the DFT domain to the image reconstruction process. As the image suggests the algorithm is successfully completed in four steps:

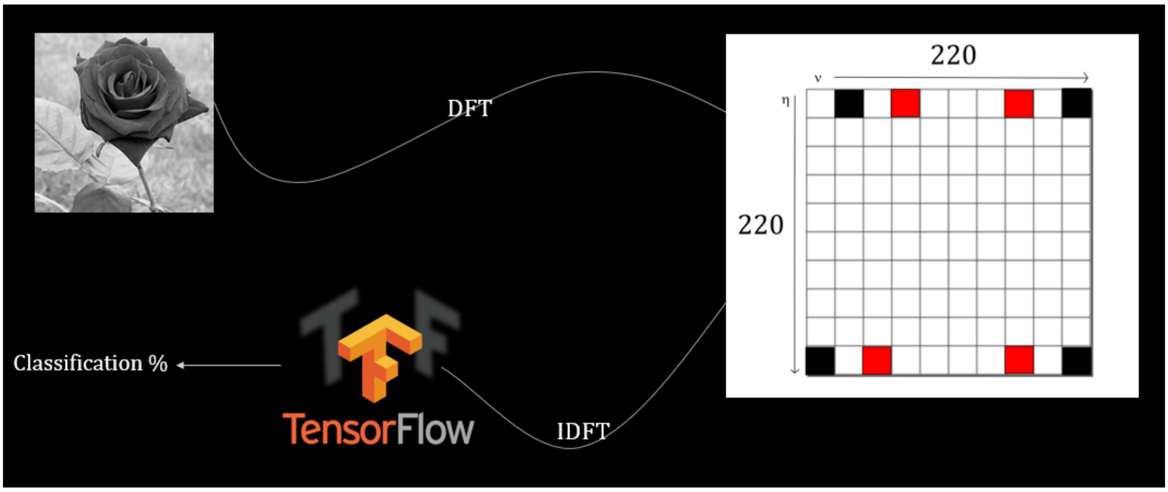


Fig. 3.5 Graphic design of the algorithm to test the relevance of every single frequency

- 1: Apply the DFT to the Figure 3.1.
- 2: In the frequency domain the algorithm removes one single frequency in each iteration until the matrix of *pixels* is totally traveled (i.e. from $\nu = \eta = 0$ to $\nu = \eta = 220$) [1]. As the reader can see in the previous image the algorithm removes 4 frequencies in each iteration (to know what is the reason for removing 4 instead of 1, see section 2.3),
- 3: Once the frequency (i.e. 4 frequencies) is removed, the algorithm applies IDFT to reconstruct the Figure 3.1.
- 4: Finally, the algorithm passes the reconstructed image to the neuronal network of Tensorflow to know what is the classification percentage without the information of the frequency (4 frequencies) that was removed in the DFT domain.

3.2.2 Frequency Filter

Besides the relevance of every single frequency in terms of information for the image reconstruction process, it is mandatory not just to analyze if each frequency is relevant but a bunch of them. Thus, the second algorithm seeks to analyze the information contained in the Fourier domain using **shells** of frequencies and not isolated frequencies.

This subsection is called *frequency filter* because it seeks to remove not just single frequencies but shells of frequencies in the DFT domain. Thus, there will be in each iteration a new $\vec{k}$ upper limit that will send to zero the frequencies that not pass the limit. For

[1] Each step in the ν and η indexes is made two by two to reduce computational time cost

instance, if $\vec{k} = 2$ the only frequencies that will be changed to zero value are $f(\nu = 0, \eta = 0)$, $f(\nu = 0, \eta = 1)$, $f(\nu = 1, \eta = 0)$ and $f(\nu = 1, \eta = 1)$ due to the fact that $\vec{k} > \sqrt{\nu^2 + \eta^2}$.

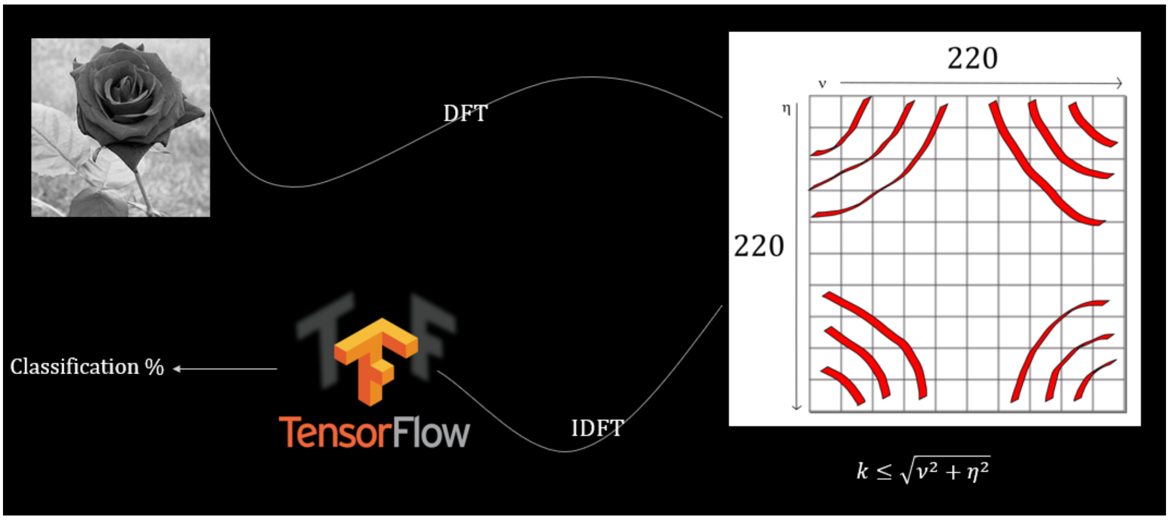


Fig. 3.6 Graphic design of the algorithm to test the relevance of frequency shells

As the image suggests the algorithm is successfully completed in four steps:

- 1: Apply the DFT to the Figure 3.1.
- 2: In the frequency domain the algorithm removes all the frequencies that satisfy the condition $\vec{k} > \sqrt{\nu^2 + \eta^2}$ given a different $\vec{k}$ in each iteration until the limit is reached (i.e. $k_{max} = \sqrt{220^2 + 220^2}$). As the reader can see in the previous image the algorithm removes 4 shells at the same time (to know what is the reason of removing 4 shells instead of 1, see section 2.3),
- 3: Once the frequency shell (i.e. 4 frequency shells) is removed, the algorithm applies IDFT to reconstruct the Figure 3.1.
- 4: Finally, the algorithm passes the reconstructed image to the neuronal network of Tensorflow to know what is the classification percentage without the information of the frequency shell (4 frequency shells) that was removed in the DFT domain.

3.2.3 Phase Annihilation

In the last two previous subsections, the algorithm has been concentrated on finding the relevance of isolated frequencies and frequency shells. However, the main concern of this document is to find the relevance of phase information not just frequency relevance. Hence, the algorithm seeks to measure how phase information is relevant to the process of digital image reconstruction.

This subsection is called *phase annihilation* due to the fact that **phase information** in the DFT domain will be removed. Although the word *remove* can be understood in a simple way, there is a special need to make a careful check of its meaning.

Phase in terms of complex numbers is an angle. Now it is useful to review section 2.4 where it was described how a complex number can be expressed in polar coordinates.

Consider the following complex number $z = Ae^{i\theta}$ where θ is its phase. If we change θ for $\theta + 2\pi$ we are with the same complex number due to the fact that $e^{\theta} = e^{\theta+2\pi}$. [2] Thus, there is a problem in the *remove* definition for phase information. For instance, if we have $\theta = 2\pi$ a succesful remove process can not be just replace that phase for $\theta = 0$ because there will not be lost of information for the reason explained previously.

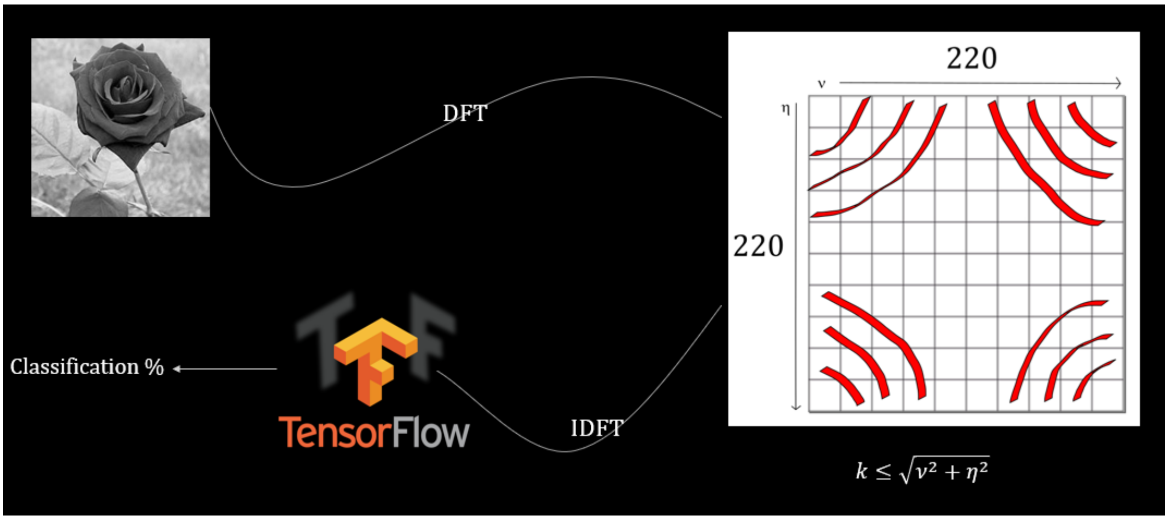


Fig. 3.7 Graphic design of the algorithm to test the relevance of phase information in frequency shells

As the reader can see, Figure 3.7 is exactly the same as Figure 3.6 because the method to annihilate phase information is the same but with a slight change: phase of each frequency that satisfies the condition over the k limit will be changed for a random number between 0 to 2π. Thus, we guarantee that phase information is being truly annihilated due to randomness.

As the image suggests the algorithm is successfully completed in four steps:

- 1: Apply the DFT to the Figure 3.1.

[2] $e^{\theta+2\pi} = e^{\theta}e^{2\pi}$, using euler's identity, $e^{\theta}e^{2\pi} = e^{\theta}(cos(2\pi) + isen(2\pi)) = e^{\theta}$

- 2: In the frequency domain the algorithm *removes* [3] the phase in all the frequency shells that satisfy the condition $\vec{k} > \sqrt{\nu^2 + \eta^2}$ given a different $\vec{k}$ in each iteration until the limit is reached (i.e. $k_{max} = \sqrt{220^2 + 220^2}$). As the reader can see in the previous image the algorithm removes phase in 4 shells at the same time (to know what is the reason of removing phase in 4 shells instead of removing phase in just one shell see section 2.3),
- 3: Once the phase information in the frequency shell (i.e. 4 frequency shells) is removed, the algorithm applies IDFT to reconstruct the Figure 3.1.
- 4: Finally, the algorithm passes the reconstructed image to the neuronal network of Tensorflow to know what is the classification percentage without the phase information of the frequency shell (4 frequency shells) that was removed in the DFT domain.

[3]The verb *removes* should be interpreted as it was explained before in the present subsection (i.e. removes does not mean change each phase for 0 value)

Chapter 4

Analysis and Results

4.1 Neural Network Recognition

4.1.1 Isolated Frequencies

The algorithm described in section 3.2.1 was executed in an approximate time of 1 hour and 30 minutes. The following plot has in the x and y axis the indexes with which the 2D matrix in the DFT domain was walked through (i.e. i and j play the role of ν and η introduced in Chapter 2); the z axis has the classification percentage of the reconstructed image that was passed to the neural network of Tensorflow without the information of the frequency with index (i, j) because it was removed [1]. The results of the execution are presented in the below plot:

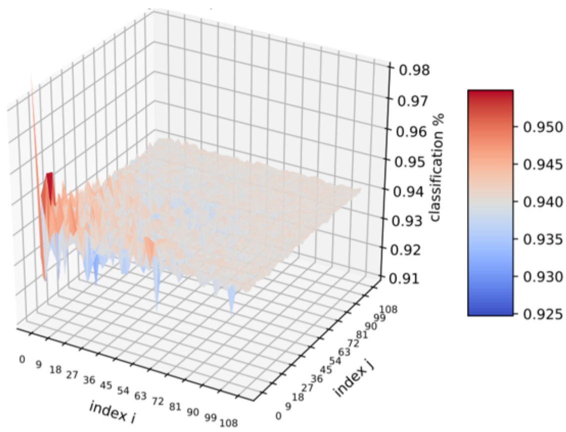


Fig. 4.1 Classification percentage due to removing process of one frequency per iteration.

[1]Section 2.3 explain why 4 frequencies are removed due to symmetry and not just one.

As it can be seen in the previous plot the lowest percentage that the neural network predicts when receives an image is 92.5% of accuracy, it means that the neuronal network does not have any trouble recognizing the image with a single frequency removed in the DFT domain because no matter which frequency is removed when the image is reconstructed and passed to Tensorflow network it predicts that Figure 3.1 is a rose, as it should be.

Hence, the relevant information content of the image to achieve a reconstruction with the lowest amount of information is not in the removing single frequency process because removing a single frequency does not disturb the image when it is reconstructed.

In light of the previous results now it is time to find where else is the critical information that will disturb the information contained in the image and will fail the recognition process made by the neuronal network. Thus, if the relevant information is not in single frequencies it is time to analyze if it is in a bunch of them, that process is called: *frequency filter*.

4.1.2 Frequency Filter

The algorithm described in section 3.2.2 was executed in an approximate time of fewer than 5 minutes. It is useful to remind that each iteration has a different $\vec{k}$ limit. The following images show how the $\vec{k}$ limit affects the amount of information that is lost:

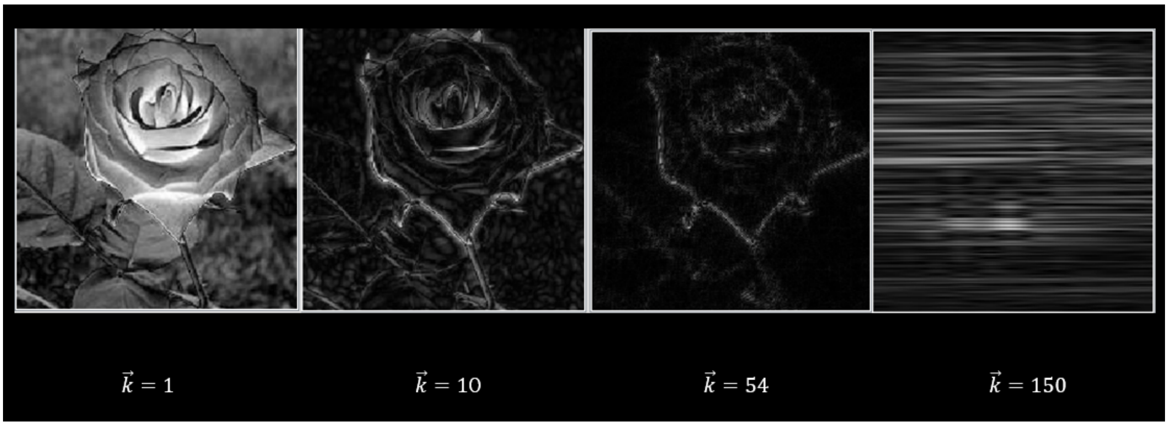


Fig. 4.2 Different $\vec{k}$ limits applied in DFT domain to Figure 3.1 to remove frequencies.

A human being would not find difficult to recognize for $\vec{k} = 1$, $\vec{k} = 10$ and, perhaps for, $\vec{k} = 54$ that the image is a rose. However, if $\vec{k}$ keeps increasing there will be a moment where the human being can not recognize the original image that was used. Thus, in this part of the algorithm now there is an important loss of information, so the crucial question is: does the neuronal network will fail to recognize the image? if so, at which $\vec{k}$ does it fail?

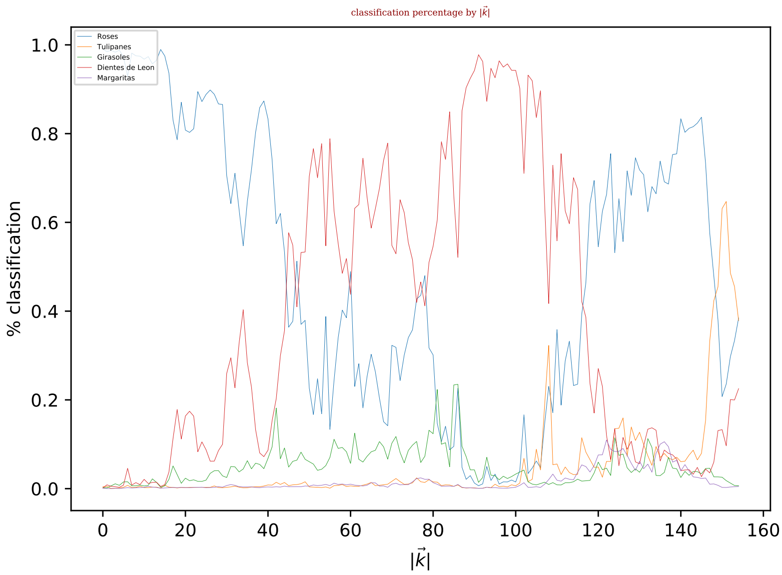


Fig. 4.3 Classification percentage predicted by the neuronal network using the frequency filter algorithm on Figure 3.1

Now it is useful to remind that there are 5 entities in the Tensorflow world: rose, tulip, sunflower, dandelion, and daisy (to see more information about how the neuronal network works, see section 1.3). Thus, the previous plot is showing what is the classification percentage of accuracy that the neuronal network predicts about each type of flower given a $\vec{k}$. As the reader can see, as $\vec{k}$ grows, the classification percentage of the rose classification curve fall off and another classification curve of a different type of flower grows, in this case, the dandelion (note that there is a point where the neuronal network begins to guess the classification process).

Hence, the main objective of this document is almost completed: **it was created a specific computational experiment condition where the neuronal network of Tensorflow is getting confused**. Thus, critical information is being handled. However, to achieve completely the main concern of this dissertation it is time to see if the relevant information that manages to confuse the neuronal network is in the phase information or not.

4.1.3 Phase Annihilation

Phase annihilation algorithm is explained in section 3.2.3 and was executed in less than 5 minutes. According to the previous subsection, there is an experimental environment where the neuronal network is getting confused in the classification process. Thus, the new algorithm seeks to find if the information that is removed from the image is relevant due to the phase or not.

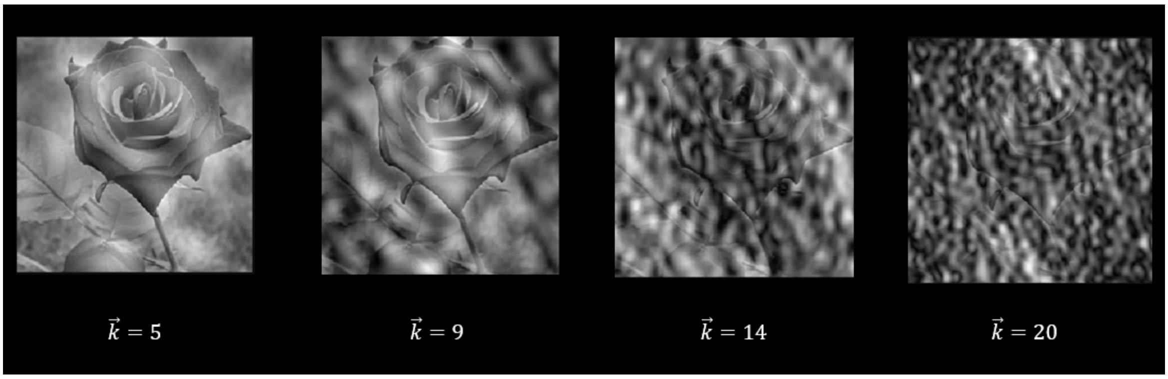
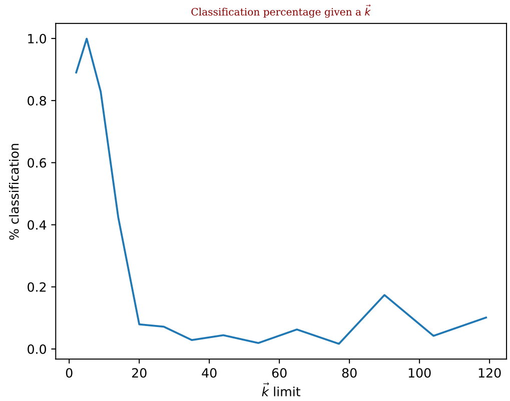


Fig. 4.4 Different $\vec{k}$ limits applied in DFT domain to Figure 3.1 to remove just phase information

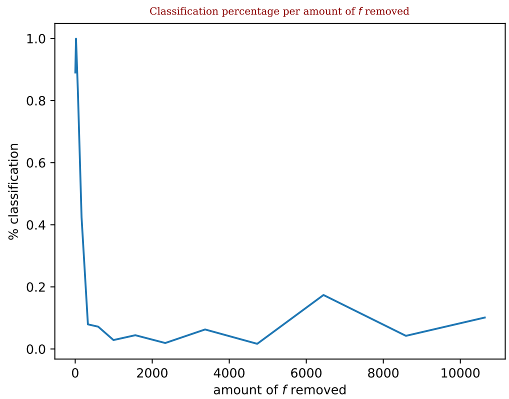
Figure 4.4 shows how the $\vec{k}$ limit affects the amount of information that is lost. The reader should compare Figure 4.2 and Figure 4.4. Notice that in Figure 4.2 in $\vec{k} = 60$ is hard (but possible) for a human being capable to recognize the rose, but now in Figure 4.4 a human being is not even capable to recognize the rose anymore if the $\vec{k}$ is approaching to 20. The reason why in *frequency filter* algorithm the $\vec{k}$ limit is 60 and in *phase annihilation* algorithm is just 20 is due to the way the information is removed.

The reader can consider in Figure 4.3 that the rose classification curve (i.e. the blue line) reaches one of its minimums almost at $\vec{k} = 60$. However, the reader should consider that in the previous algorithm the complete frequency was removed (i.e. phase and magnitude) because the complete complex number was sent to zero value. Now, the phase annihilation algorithm will do the same process as the previous algorithm but instead of sending to zero the whole complex number it will just *remove* the phase (*remove* should not be understood as sending phase information to zero, to see the explanation about how to remove the phase of a complex number see section 3.2.3).

In Figure 4.5 (a) the curve is reaching one of the minimums nearly $\vec{k} = 20$. What this information does it mean? In frequency filter algorithm the neuronal network gets totally confused with almost a $\vec{k} = 60$ because the classification percentage that the neuronal network gives of the image being a rose was almost a 20%. However, it is important to remind that frequency filter removes (i.e. send to zero the complex number) all the frequencies that not satisfy the condition over the $\vec{k}$ limit. But, in phase annihilation the algorithm replaces the phase for a random number letting amplitude unmodified, so the algorithm ensures that just



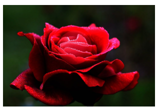
(a) Classification percentage removing phase information given a $\vec{k}$ limit



(b) Classification percentage according to the amount of $f$recuencies changed

Fig. 4.5 Classification percentage of accuracy changing phase information on Figure 3.1

phase is being removed. *Removing* phase information shows that the neuronal network is getting totally confused at $\vec{k} = 20$ which means that now the algorithm is changing less but certainly more crucial information.



(a) Image 1

(b) Image 2

(c) Image 3

(d) Image 4

(e) Image 5

Fig. 4.6 Set of images where the annihilation algorithm will be executed

Another way to see the last conclusion is seeing Figure 4.5 (b). This plot says that the neuronal network fails in the classification process when almost 1000 frequencies (just phase of these frequencies) are being manipulated over the 12100 in total [2] it means an 8.26%. In frequency filter algorithm the neuronal network fails when almost 3000 frequencies are being changed over a 24.79%. Thus, the computational experiment let conclude about the main concern of this dissertation, **removing phase information rather than frequency**

[2]Total number of frequencies is 48400. However, due to symmetry (see section 2.3) there is only need to work with 12100 and apply the algorithm as a mirror to the rest of the frequencies

information affects almost 3 times more the information content of the image for a digital image reconstruction process.

As a final step, the previous conclusion can just apply to Figure 3.1, perhaps it does not apply to another set of images. Hence, the phase annihilation algorithm is applied to the set of images of roses (Figure 4.6) to see if the conclusion still holds.

As the reader can see in Figure 4.7 the conclusion about the low $\vec{k}$ limit that is needed to let the neural network get confused manipulating phase information in the image holds in the set of images of roses.

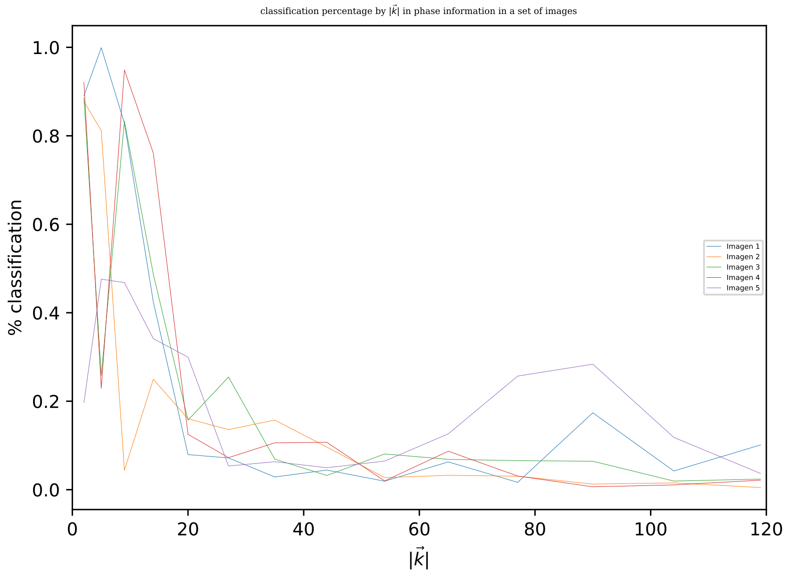
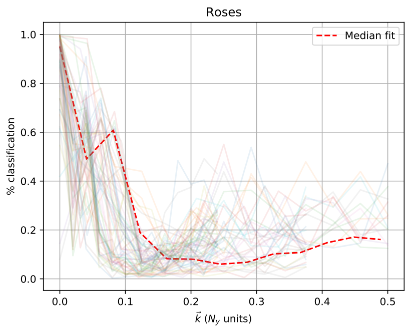
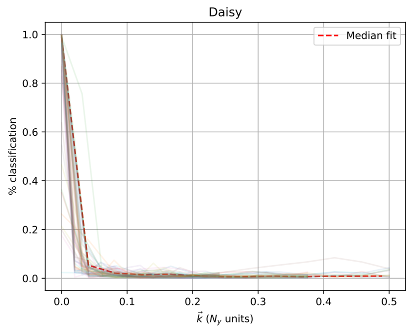


Fig. 4.7 Classification percentage removing phase information in the set of roses given a $\vec{k}$ limit

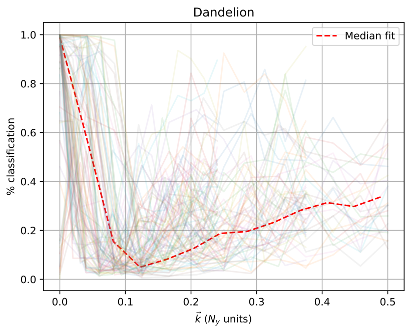
Finally, this conclusion must be extended to a set of images including another type of flowers to be general because it can only apply to roses. Thus, the following plots are phase annihilation algorithm applied to 5 different sets of images: tulips, roses, sunflowers, dandelion, and daisies. These sets were taken from the data that the ANN used for the training process. 10% of the images of each type of flower was saved apart until the end of this computational experiment to test the ANN process with images that never saw before. This set of images is usually called the validation set. For this case, 10% is 641 roses images, 633 daisies images, 898 dandelions images, 799 sunflowers images, and 799 tulips images.



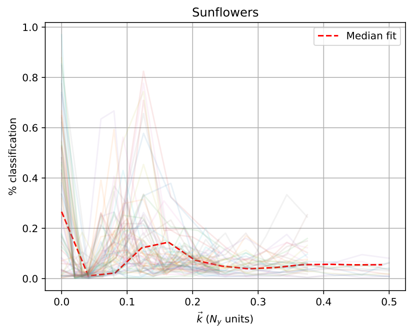
(a) Roses



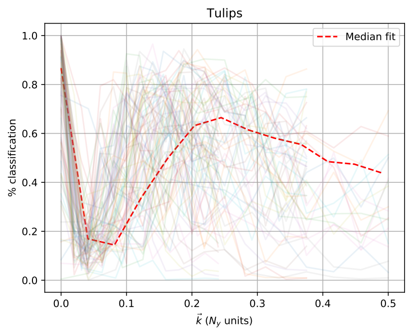
(b) Daisies



(c) Dandelions



(d) Sunflowers



(e) Tulips

Fig. 4.8 Classification percentage removing phase information in the set of images given a $\vec{k}$ limit in f_{Ny} units.

Without removing phase information (i.e. $\vec{k} = 0\ f_{Ny}$ where f_{Ny} is Nyquist frequency units [3]) the accuracy of the ANN is almost 100% (except for the case of sunflowers that probably is due to the training test of this type of flowers was not good enough). When phase information is removed with $\vec{k}$ between $0.0\ f_{Ny}$ and $0.1\ f_{Ny}$ the results are very different, but the average result is that the ANN accuracy falls (in daisies is a fast fall but in dandelions is a slow fall). However, the conclusion about phase information found for the Figure 3.1 and then for the set of roses 4.6 is also valid for the general case of more types of flowers as the reader can see in Figure 4.8 since all ANN classification percentages above $\vec{k} = 0.1\ f_{Ny}$ fall down sharply (i.e. the ANN is not able to recognize the image with the lack of phase information for $\vec{k} = 0.1\ f_{Ny}$).

As a further research it would be interesting analyze why in plots like Fig 4.8 (c) and (e) the classification percentage after having a strong fall for $\vec{k} = 0.1\ f_{Ny}$ the percentage rises again.

[3] We use Nyquist frequency units because each image has a different size and the $\vec{k}$ limit is not comparable between them at least the limit is in Nyquist frequency units.

Chapter 5

Conclusions

A preliminary study of the information of an image is made to see what is the information available in an image. Then, *isolated frequencies* algorithm is applied where complex values of single frequencies are removed to see if the neural network fail the recognition that is no the case. Next, *frequency filter* algorithm is executed where shells of complex values are removed and in this case, the neural network starts failing the recognition at some $\vec{k}$. Despite the main purpose of this document is studying phase information in digital images the study of the complete complex value is made to be able to make a comparison between the complete information against phase information. Finally, *phase annihilation* algorithm is applied to three cases: a rose image, a set of roses and multiple sets of different type of flowers. All cases show that the $\vec{k}$ limit in phase information is several times lower than the $\vec{k}$ limit in complete information to achieve the neural network reduces its accuracy. It means that removing less phase information is needed to fail the recognition process made by the neural network.

We find that the norm of complex numbers of the DFT of an image is larger around middle frequencies. This is a known result that is the base of the compressive processes such as JPEG compression where high frequencies are discarded due to the less relevance in terms of information (in Chapter 3 is shown that low frequencies are less than 1% and high frequencies are just 3% of the total frequencies on the images).

We also find that the magnitude values plot has a well defined shape, while the phase values plot looks like a random plot. For this reason, magnitude is usually chosen over phase for digital image processes (e.g. compression or reconstruction) due to the intuitive perception. Nevertheless, phase information is more relevant in terms of the amount of information as it is shown in Chapter 4.

Isolated frequencies algorithm (where complex values of single frequencies are removed) set the conclusion that in the frequency domain the relevance of single complex values is negligible due to the high classification percentages predicted by the neuronal network as it can be seen in Figure 4.1.

Frequency filter sets that removing shells of complex values can make the neural network reduces its recognition accuracy. It means that there is a point where lack of complete information impacts the classification. We find that frequencies below $\vec{k} = 0.27 f_{Ny}$ (*Ny* means Nyquist frequency) are required to have classification percentages above 99%.

We also used a filter where only phase information is shuffled called *phase annihilation* algorithm where in comparison to *frequency filter* algorithm the complete complex value (i.e. amplitude and phase) is destroyed. We find that manipulating the phase information has a strong impact on classification process. The amount of phase information removed is extended until $\vec{k} = 0.09 f_{Ny}$ where the neural network predicted a classification accuracy above 99%. Hence, phase information is more relevant than complete information (magnitude and phase) for a digital image reconstruction process due to the fact that fewer phase information is needed to fail the classification process of the neural network.

This previous conclusion is not just appliable for Fig 3.1 or the set of roses Fig 4.6 but for a multiple set of images of different kind of flowers. This conclusion is tested over the 10% of the total images in the neural network set. We find that the classification percentage of the neural network after removing phase information greater than $\vec{k} = 0.1\ f_{Ny}$ in multiples images is less than 50% which shows that phase information has a relevance in terms of information content for a digital image.

We conclude that images are harder to classify when phase information is removed than amplitude information. In other words, phase information has more relevance in terms of information than amplitude information.

References

[1] José van Dijck. Digital photography: communication, identity, memory. *Visual Communication*, 7(1):57–76, February 2008.

[2] J. R. Fienup. Iterative Method Applied To Image Reconstruction And To Computer-Generated Holograms. *Optical Engineering*, 19(3), June 1980.

[3] J. Fridrich. Applications of data hiding in digital images. volume 1, page 9, Brisbane, Qld., Australia, 1999. Queensland Univ. Technol.

[4] V. V.D. Shah. Image Processing and its Military Applications. *Defence Science Journal*, 37(4):457–468, January 1987.

[5] Matteo Stocchetti. Images and Power in the Digital Age: The political role of digital visuality. *KOME*, 2(2), 2014.

[6] C J Vyborny and M L Giger. Computer vision and artificial intelligence in mammography. *American Journal of Roentgenology*, 162(3):699–708, March 1994.

[7] E.J. Candes and M.B. Wakin. An Introduction To Compressive Sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, March 2008.

[8] M. Hayes, Jae Lim, and A. Oppenheim. Signal reconstruction from phase or magnitude. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(6):672–680, December 1980.

[9] A.V. Oppenheim and J.S. Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 1981.

[10] A. Oppenheim, Jae Lim, G. Kopec, and S. Pohlig. Phase in speech and pictures. volume 4, pages 632–637, Washington, DC, USA, 1979. Institute of Electrical and Electronics Engineers.

[11] Yoav Shechtman, Yonina C. Eldar, Oren Cohen, Henry Nicholas Chapman, Jianwei Miao, and Mordechai Segev. Phase Retrieval with Application to Optical Imaging: A contemporary overview. *IEEE Signal Processing Magazine*, 32(3):87–109, May 2015.

[12] E.J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, February 2006.

[13] Spiros Chountasis, Vasilios N. Katsikis, and Dimitrios Pappas. Digital Image Reconstruction in the Spectral Domain Utilizing the Moore-Penrose Inverse. *Mathematical Problems in Engineering*, 2010:1–14, 2010.

[14] R.C. Puetter, T.R. Gosnell, and Amos Yahil. Digital Image Reconstruction: Deblurring and Denoising. *Annual Review of Astronomy and Astrophysics*, 43(1):139–194, September 2005.

[15] Material del curso de Métodos Computacionales de Uniandes: ComputoCienciasUniandes/MetodosComputacionales, November 2018. original-date: 2013-01-06T03:28:57Z.

[16] Dave Anderson and George McNeill. Artificial neural networks technology. *Kaman Sciences Corporation*, 258(6):1–83, 1992.

[17] Sam Abrahams, Danijar Hafner, Erik Erwitt, and Ariel Scarpinelli. *TensorFlow for machine intelligence: a hands-on introduction to learning algorithms*. 2016. OCLC: 959836902.

[18] Rafael C. Gonzalez and Richard E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J, 3rd ed edition, 2008.

[19] L. Rabiner. On the use of symmetry in FFT computation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(3):233–239, June 1979.

[20] J. Urrea Nino. Evaluation of phase retrieval in a compressive computational ghost imaging setup. *Undergraduate thesis dissertation*, Uniandes, 2017.