



UNIVERSITÀ
DEGLI STUDI
DI PADOVA



Università degli Studi di Padova

DEPARTMENT OF INFORMATION ENGINEERING

Master's Degree in Bioengineering

**Smartphone-based retinal image analysis:
a convolutional neural network approach
for automatic cataract detection**

July 22, 2020

Candidate:

Alberto Peraro

Supervisor:

Prof.ssa Maria Pia SACCOMANI

Co-supervisor:

Ing. Fabio SCARPA

*This work is supported by:
D-EYE S.r.l., Padova
www.d-eyecare.com*





Ad maiora, semper.

Contents

Abstract	7
1 Introduction and background	8
1.1 Eye anatomy	8
1.1.1 Retina	10
1.1.2 Lens anaotmy	12
1.2 Cataract	14
1.2.1 Risk factors	14
1.2.2 Cataract types	14
1.2.3 Surgery	17
1.2.4 Diagnosis	18
2 Diagnostic tools for ophthalmology	21
2.1 Slit lamp	21
2.1.1 Direct focal illumination	22
2.2 Ophthalmoscope	24
2.2.1 Direct ophthalmoscope	24
2.2.2 Indirect ophthalmoscope	25
2.3 Fundus camera	27
2.4 D-EYE	29
3 Deep Learning	32
3.1 Artificial Neural Network (ANN)	32
3.2 Convolutional Neural Network (CNN)	38
3.2.1 Convolutional layer	41
3.2.2 ReLU layer	42
3.2.3 Pooling layer	42
3.2.4 Fully connected layer	43
3.2.5 Output layer	43
3.3 Training phase	44
3.3.1 Overfitting and underfitting	45
3.3.2 Early stopping and drop out	47
3.3.3 Data augmentation	48
3.3.4 Transfer learning	49
3.4 Deep learning in Biomedical imaging	50
3.4.1 History of ML in computer vision and medical imaging fields	50
3.4.2 Applications	51
3.4.3 Focus on ophthalmology	53
3.5 Related work	56

4	Method	60
4.1	Dataset	60
4.1.1	Dataset 1 - ODIR5k and Retina dataset	61
4.1.2	Dataset 2 - DEYE-like Tool	65
4.2	Pre-processing	66
4.2.1	Fourier Transform	67
4.2.2	Green channel	68
4.2.3	Histogram equalization	69
4.3	CNN architecture	70
4.4	CNN training	74
4.4.1	Data augmentation	77
4.5	CNN testing	77
5	Results	80
5.1	Dataset 1	82
5.2	Dataset 2	82
5.3	Post-processing	84
6	Conclusion	88
	Bibliography	89
	Appendix 1	95
	Acknowledgements	99

Abstract

Cataract, the clouding of the crystalline lens that focuses the light entering the eye onto the retina, is one of the most serious eye disease leading to blindness. Early detection and treatment can reduce the rate of complications in cataract patients. This is especially relevant in developing countries where access to healthcare is poor and the lack of eye specialist makes this diagnosis really hard.

In this context D-EYE emerges as a smartphone-based ophthalmoscope aims to be efficiently used both by ophthalmologists in clinics, for large screening or in rural areas by not medical personnel. The strength of this device are the possibility to automatically perform diagnosis and the capability of recording and transmitting high-definition images and videos of the fundus oculi.

In order to extend the possibilities concerning D-EYE, this project focuses on the development of an algorithm able to automatically detect cataract through retinal images. Several recent studies in literature suggest to use convolutional neural network as a possible solution to this task. For this reason the proposed algorithm is based on MATLAB (The Mathworks Inc., Natick, MA, USA) and in particular a custom convolutional neural network has been implemented using "Deep Learning Toolbox". After an iterative process of refining where different strategies were tested to achieve the best performances, finally the CNN obtains promising results. In terms of classification percentages the leading network successfully classifies 95.9% of the fundus images analysed.

The following part of this thesis is organized as follow. Chapter 1 introduces eye anatomy with particular regard to crystalline lens and the characteristics of the target disease: cataract. Chapter 2 describes different ophthalmological instruments related to cataract diagnosis including D-EYE. Chapter 3 presents an overview of deep learning and convolutional neural networks. Chapter 4 describes in detail the methods utilized. Chapter 5 regards findings and some discussions about the values achieved. The conclusion of the thesis is given in chapter 6.

1. Introduction and background

In modern society everyone spend a consistent part of his daytime in looking at some electronic device, either for work, for communicate or just for fun. For this reason the demand on using eyes is growing more and more. Eye protection and in general eye research become increasingly important for all the population.

Cataract, diabetic retinopathy, conjunctivitis, glaucoma are some of the most common eye diseases. According to the WHO, cataract is one of the leading causes of visual impairment worldwide and one of the main causes of blindness. The NEI (National Eye Institute), one of 27 institutes and centers of the US National Institutes of Health, affirms that by 2050, the number of people in the U.S. with cataract is expected to reach up to about 50 million. Muller-Breitenkamp et al. present some other forecasts with an estimation of 40 million of people who will suffer from cataract in 5 years [1]. In low and middle income countries and regions, these figures are even higher because of lower investment in health. Besides, the longer a patient has an untreated cataract, the more severe is the vision impairment. Although it is well known that more early diagnosis and treatment can reduce the suffering of cataract patients and prevent visual impairment from turning into blindness, there is still a lack of timely treatment in less developed areas because of deficient skilled opthalmologists and poor eye care services. In a recent article [2] the authors try to estimate regional and global cataract prevalence, its prevalence in different age groups, and the determinants of heterogeneity. For this reason they use international databases such as PubMed, Web of Science, Scopus, Embase, and other sources of information to conduct a systematic search for all articles concerning the prevalence of age-related cataract and its types in different age groups. Of the 9922 identified articles, 45 studies with a sample size of 161947 were included in the analysis, and most of them were from the Office for the Western Pacific Region. At the end of this paper the conclusion reveals that from the public health point of view, cataract is still a global challenge, especially in Western Pacific countries. Despite the lack of inter-gender differences, cataract prevalence increases with age, especially after the age of 60 years. Knowledge about cataract prevalence can support health-care planners in planning and prioritizing resource allocation.

1.1. Eye anatomy

To understand the diseases and conditions that can affect the eye, it helps to understand basic eye anatomy. The human eye (figure 1) measures approximately 22 to 27 mm in anteroposterior diameter and 69 to 85 mm in circumference. The human eyeball consists of three primary layers, with each of them being subdividable. The three primary layers and their respective subdivisions include: (1) the outermost supporting layer of the eye, which consists of clear cornea, opaque sclera, and their zone of interdigitation, designated as the limbus; (2) the middle uveal layer of the eye, constituting the central vascular layer of the globe, which encompasses the iris, ciliary body, and choroid; and (3) the interior layer of the eye, commonly designated as the retina [3].

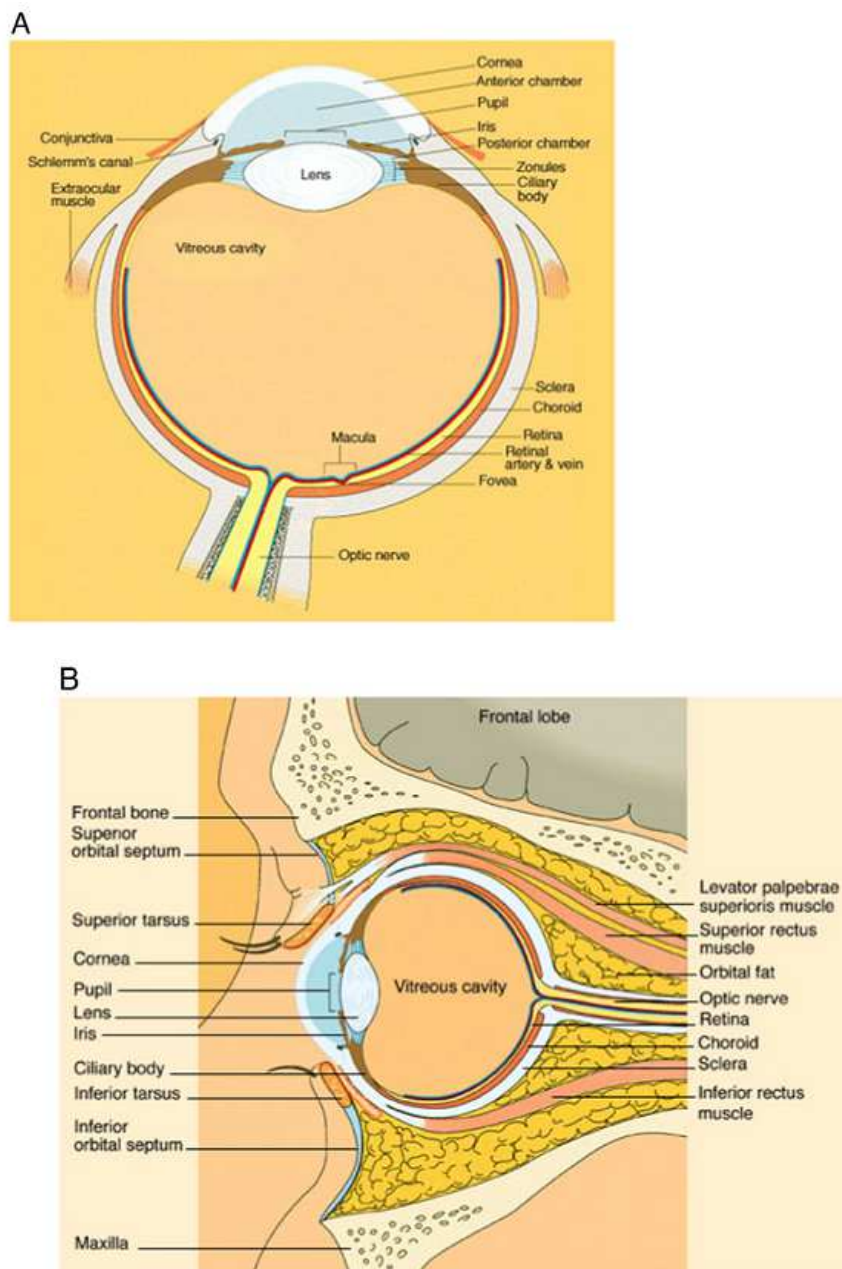


Figure 1. Panel A, Cross section of the eye. Panel B, Sagittal section of the eye in orbit.

Here the eye will be presented starting from the outside, going in through the front and working to the back. The eye sits in a protective bony socket called the orbit. Six extraocular muscles in the orbit are attached to the eye. These muscles move the eye up and down and side to side, and rotate the eye. The extraocular muscles are attached to the white part of the eye, called sclera.

This is a strong layer of tissue that covers nearly the entire surface of the eyeball. The surface of the eye and the inner surface of the eyelids are covered with a clear membrane called conjunctiva. Tears lubricate the eye and are made up of three layers. These three layers together are called the tear film. The eye's lacrimal gland sits under the outside edge of the eyebrow (away from the nose) in the orbit. This gland produces the watery part of the tears. The meibomian gland makes the oil that becomes another part of the tear film. Tears drain from the eye through the tear duct.

Light is focused into the eye through the clear, dome-shaped front portion of the eye called cornea. Behind the cornea is a fluid-filled space called anterior chamber. The fluid is called aqueous humor and it is mainly made of water with the presence of hyaluronic acid. The eye is always producing aqueous humor. To maintain a constant eye pressure, aqueous humor also drains from the eye in an area called the drainage angle. Behind the anterior chamber is the eye's iris (the colored part of the eye) and the dark hole in the middle called pupil. Muscles in the iris dilate or constrict the pupil to control the amount of light reaching the back of the eye. Directly behind the pupil sits the lens. The lens focuses light toward the back of the eye. The lens changes shape to help the eye focus on objects up close. Small fibers called zonules are attached to the capsule holding the lens, suspending it from the eye wall. The lens is surrounded by the lens capsule, which is left in place when the lens is removed during cataract surgery. A replacement intraocular lens goes inside the capsule, where the natural lens was. The vitreous cavity lies between the lens and the back of the eye. A jellylike substance called vitreous humor fills the cavity, nourishing the inside of the eye and helping the eye hold its shape.

Light that is focused into the eye by the cornea and lens passes through the vitreous onto the retina — the light-sensitive tissue lining the back of the eye. A tiny but very specialized area of the retina, called macula, is responsible for giving detailed, central vision. The other part of the retina, the peripheral retina, provides with peripheral (side) vision. The retina has special cells called photoreceptors. These cells change light into energy that is transmitted to the brain. There are two types of photoreceptors: rods and cones. Rods perceive black and white, and enable night vision. Cones perceive color, and provide central (detail) vision. The retina sends light as electrical impulses through the optic nerve to the brain. The optic nerve is made up of millions of nerve fibers that transmit these impulses to the visual cortex — the part of the brain responsible for the sight.

1.1.1. Retina

When an ophthalmologist uses an ophthalmoscope to look into the eye, he sees the following view of the retina (figure 2). In the center of the retina is the optic nerve, a circular to oval white area measuring about 2 x 1.5 mm across. From the center of the optic nerve radiate the major blood vessels of the retina. Approximately 17 degrees (4.5-5 mm), or two and half disc diameters to the left of the disc, can be seen the slightly oval-shaped, blood vessel-free reddish spot, the fovea, which is at the center of the macula.

A circular field of approximately 6 mm around the fovea is considered the central retina while, beyond this, is peripheral retina stretching to the ora ser-

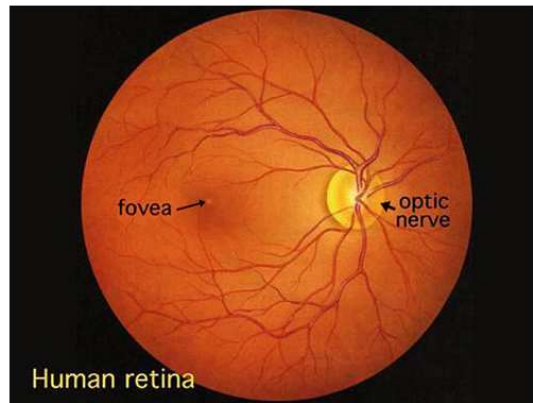


Figure 2. A view of the retina seen through an ophthalmoscope.

rata, 21 mm from the center of the retina (fovea). The total retina is a circular disc of diameter between 30 and 40 mm. The retina is approximately 0.5 mm thick and lines the back of the eye. The optic nerve contains the ganglion cell axons running to the brain and, additionally, incoming blood vessels that open into the retina to vascularize the retinal layers and neurons (figure 3). A radial section of a portion of the retina reveals that the ganglion cells (the output neurons of the retina) lie innermost in the retina closest to the lens and front of the eye, and the photoreceptors (the rods and cones) lie outermost in the retina against the pigment epithelium and choroid. Light must, therefore, travel through the thickness of the retina before striking and activating the rods and cones (figure 3). Subsequently the absorption of photons by the visual pigment of the photoreceptors is translated into first a biochemical message and then an electrical message that can stimulate all the succeeding neurons of the retina. The retinal message concerning the photic input and some preliminary organization of the visual image into several forms of sensation are transmitted to the brain from the spiking discharge pattern of the ganglion cells.

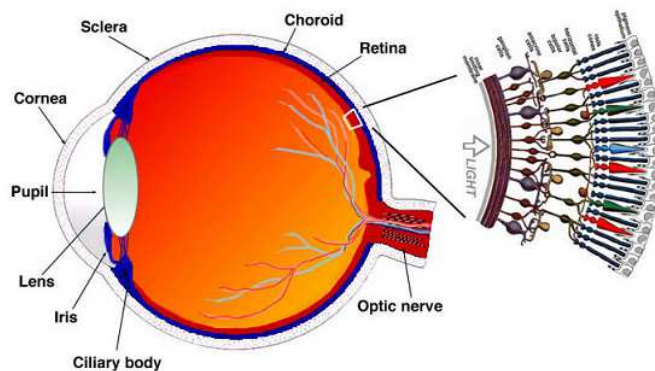


Figure 3. A drawing of a section through the human eye with a schematic enlargement of the retina.

A simplistic wiring diagram of the retina emphasizes only the sensory photoreceptors and the ganglion cells with a few interneurons connecting the two cell types such as seen in figure 4.

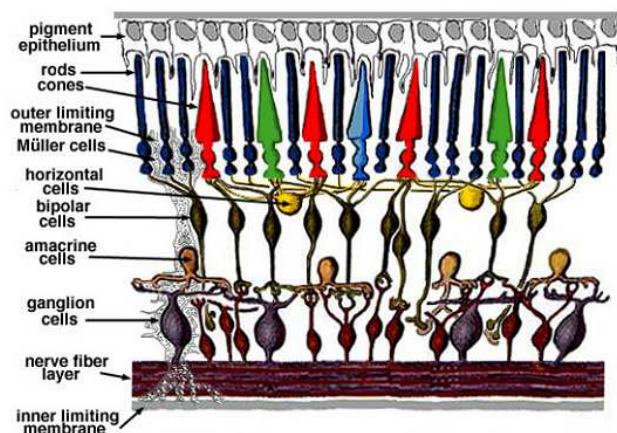


Figure 4. Simple diagram of the organization of the retina.

When an anatomist takes a vertical section of the retina and processes it for microscopic examination it becomes obvious that the retina is much more complex and contains many more nerve cell types than the simplistic scheme (above) had indicated. It is immediately obvious that there are many interneurons packed into the central part of the section of retina intervening between the photoreceptors and the ganglion cells.

All vertebrate retinas are composed of three layers of nerve cell bodies and two layers of synapses. The outer nuclear layer contains cell bodies of the rods and cones, the inner nuclear layer contains cell bodies of the bipolar, horizontal and amacrine cells and the ganglion cell layer contains cell bodies of ganglion cells and displaced amacrine cells. Dividing these nerve cell layers are two neuropils where synaptic contacts occur.

The first area of neuropil is the outer plexiform layer (OPL) where connections between rod and cones, and vertically running bipolar cells and horizontally oriented horizontal cells occur.

The second neuropil of the retina, is the inner plexiform layer (IPL), and it functions as a relay station for the vertical-information-carrying nerve cells, the bipolar cells, to connect to ganglion cells. In addition, different varieties of horizontally- and vertically-directed amacrine cells, interact in further networks to influence and integrate the ganglion cell signals. It is at the culmination of all this neural processing in the IPL that the message concerning the visual image is transmitted through ganglion cells to the brain along the optic nerve [4].

1.1.2. Lens anatomy

Going deeper in the anatomy of the lenses and in their transparency is useful to better understand the pathophysiology of their related diseases. The lens is a transparent structure that is devoid of any blood supply. Anteriorly, the lens

surface is covered by a monolayer of epithelial cells. In addition to maintaining lens metabolic activity, epithelial cells replicate to produce daughter cells, which migrate and differentiate into fiber cells. Lens fiber cells make up greater than 95% of the lens and are stretched out to form compact, concentric layers (“shells”), thereby reducing intercellular space (figure 5). Superficial lens fibers are nucleated and are metabolically active while deeper fibers, which make up most of the lens, are organelle-free with minimal metabolic activity. Interiorly, fiber cells have a high expression of soluble crystallin proteins but are devoid of nuclei, mitochondria, endoplasm reticulum, ribosomes, and other organelles. Lens crystallins make up almost 90% of proteins in the mature lens. In humans, the non-nucleated human lens fiber cells consist of α -crystallins, β -crystallins and γ -crystallins.

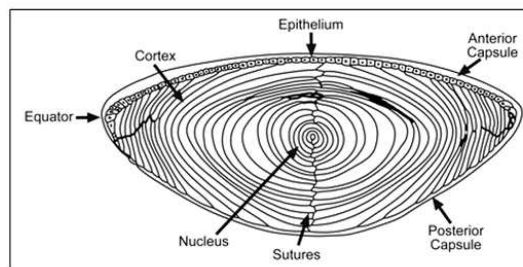


Figure 5. Schematic presentation of the cross-sectional view of mammal.

As said before, lens is essential for focusing light onto the retina and it can perform this function due to its transparent and dioptric properties. Transparency of the lens depends on avascularity, the paucity of organelles, narrow inter-fiber spaces, and regular organization of cells and proteins. At the cellular level, there is a limited light-scattering by organelles due to their limited presence in the lens. Moreover, organelles are located away from the light path, exiled to the equator in the fibers from the central epithelium, thereby reducing light scattering in the lens. Transparency is also achieved by the short-range spatial order of proteins. In fiber cells, crystallins are densely packed in a short-range order of about 250–400 mg/mL. The small protein size (<10 nm diameter), together with the close packing at high concentration, renders their wavelength less than that of light. Furthermore, dense packing of protein aggregates reduces fluctuations of protein density and reduces the refractive index below wavelength of light. Protein crystallization and precipitation are further deterred through a specialized mixture of crystallin protein forms (α , β and γ form), which confer superior solubility and native protein conformations in the lens. In addition to their structural function within the lens, by increasing the refractive index, β - and γ -crystallins exhibit high solubility and thermodynamic stability to prevent scattering of light. The α -crystallins serve as chaperones by partially binding to denatured proteins within the lens cells to form high-molecular-weight aggregates that maintain protein solubility and transparency.

In the cortex of the lens, transparency is enhanced by a high spatial order of fiber architecture with narrow intercellular spaces, which then compensates for light-scattering due to refractive index differences between membranes and cytoplasm. In the nucleus, high spatial order is not required due to minimal light scattering and negligible differences in the refractive index between fiber

membranes and cytoplasm. The cornea traps light with wavelength below 310 nm. Interestingly, the mammalian lens possesses small-molecular-weight UV filters such as tryptophan metabolites that remove UV radiation between 300–400 nm [5].

1.2. Cataract

A cataract is a clouding of the natural intraocular crystalline lens that focuses the light entering the eye onto the retina. This cloudiness can cause a decrease in vision and may lead to eventual blindness if left untreated. As new lens fibres continue to be laid down in the crystalline lens, and existing ones are not replaced, the lens is unusual in being one of the few structures of the body that continues to grow during life. The transparency of the lens is maintained by many interdependent factors that are responsible for its optical homogeneity, including its microscopic structure and chemical constituents. With ageing, there is a gradual accumulation of yellow-brown pigment within the lens, which reduces light transmission. There are also structural changes to the lens fibres, which result in disruption of the regular architecture and arrangement of the fibres that are necessary to maintain optical clarity [6]. Cataracts often develop slowly and painlessly, so vision and lifestyle can be affected without a person realizing it. In particular vision meets a gradual decline, which cannot be corrected with glasses. As there are a wide variety of cataract types, there is a large spectrum of visual symptoms associated with cataractous changes. Common complaints include blurry vision, difficulty reading in dim light, poor vision at night, glare and halos around lights, and occasionally double vision. Other signs of cataracts include frequent changes in the prescription of glasses, loss of contrast sensitivity, loss of ability to discern colors and a new ability to read without reading glasses in patients over 55 (the so called "second sight" phenomenon).

1.2.1. Risk factors

Taking into account risk factors, the ones associated with cataract formation vary with socioeconomic and geographical differences. In the developing world a multitude of factors, such as malnutrition, acute dehydrating diseases at young age, exposure to excessive ultraviolet rays, smoking and steroid use seem to be important. In many developing countries cataracts are common in young adults, frequently associated with atopic disorders and their treatment as well as with diabetes or elevated blood sugar. Other causes of cataract include trauma in a variety of forms (direct penetration, contusion, radiation, electrical, or metabolic) and congenital disorders.

1.2.2. Cataract types

There are several types of cataract. They can be divided mainly based on causes, symptoms or anatomical regions where they occur. Since, as said before, ageing

is one of the most relevant causes for cataract the description will start from the types of age-related cataracts: nuclear sclerotic, cortical, and posterior subcapsular. As a person ages, any one type, or a combination of any of these three types, can develop over time.

Nuclear sclerotic cataract (figure 6) is the most common type of age-related cataract, caused primarily by the hardening and yellowing of the lens over time. "Nuclear" refers to the gradual clouding of the central portion of the lens, called the nucleus; "sclerotic" refers to the hardening, or sclerosis, of the lens nucleus. As this type of cataract progresses, it changes the eye's ability to focus, and close-up vision (for reading or other types of close work) may temporarily improve. This symptom is referred to as "second sight," but the vision improvement it produces is not permanent. A nuclear sclerotic cataract progresses slowly and may require many years of gradual development before it begins to affect vision.

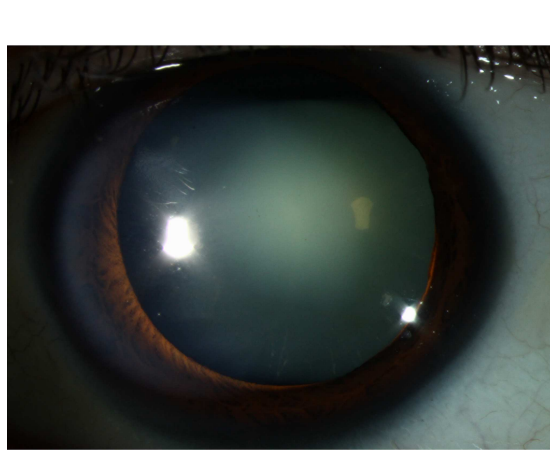


Figure 6. Eye affected by nuclear sclerotic cataract.

Cortical cataracts: "cortical" refers to white opacities, or cloudy areas, that develop in the lens cortex, which is the peripheral (outside) edge of the lens. Changes in the water content of the lens fibers create clefts, or fissures, that look like the spokes of a wheel pointing from the outside edge of the lens in toward the center. These fissures can cause the light that enters the eye to scatter, creating problems with blurred vision, glare, contrast, and depth perception. People with diabetes are at risk for developing cortical cataracts.

Posterior subcapsular cataracts begins as a small opaque or cloudy area on the "posterior," or back surface of the lens. It is called "subcapsular" because it forms below the lens capsule, which is a small "sac", or membrane, that encloses the lens and holds it in place. Subcapsular cataracts can interfere with reading and create "halo" effects and glare around lights. People who use steroids, or have diabetes, extreme nearsightedness, and/or retinitis pigmentosa may develop this type of cataract. Subcapsular cataracts can develop rapidly and symptoms can become noticeable within months.

Besides these age-related types it is possible to meet, for example, congenital cataracts, traumatic cataracts, secondary cataracts and radiation cataracts.

Congenital ones are cataracts you are born with or that form when you are a child. Some are linked to your genes, and others are due to an illness, like rubella, that your mother had during pregnancy. When they are small or outside the center of the lens, they may not need treatment. But when a baby is born with one that blocks vision, a doctor needs to remove it because it can stop the eye from learning to see.

Many kinds of injuries can lead to a cataract. You can get one if you are hit in the eye by a ball or get hurt from a burn, chemical, or splinter. In these cases physicians talk about **traumatic cataracts**. The disease could come on soon after the injury or not show up until years later.

When another condition or a medical treatment leads to a cataract, doctors call it **secondary**. Diabetes, taking steroids (like prednisone), and even cataract surgery are possible causes. When you get one after cataract surgery, it is called a posterior capsule opacification (PCO). Doctors can treat it with a quick procedure called YAG laser capsulotomy.

UV rays from the sun can be a significant cause of cataracts. People who spend a lot of time outside, or who have received radiation treatment for cancer, could develop cataracts as a result.

Furthermore some other types of cataract exist and they are categorized according to the particular shape of the disease or if there are present some peculiar signs. This category includes: **lamellar or zonular cataracts**, christmas tree cataracts, brunescant cataracts and diabetic snowflake cataracts. In order, the first ones typically shows up in younger children and in both eyes. The genes that cause them are passed from parent to child. They are so called because form fine white dots in the middle of the lens and may take on a Y shape (figure 7). Over time, the whole center of the lens may turn white.

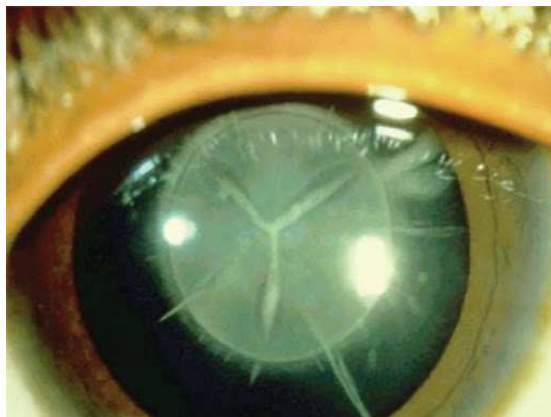


Figure 7. Eye affected by lamellar/zonular cataract.

Christmas tree or polychromatic cataracts, they form shiny, colored crystals in your lens (figure 8). They are most common in people who have a condition called myotonic dystrophy.

If you do not treat a nuclear cataract, it turns very hard and brown. When that happens, it is called **brunescant**. This kind of cataract makes it hard for you to tell colors apart, especially blues and purples. Surgery to remove it is

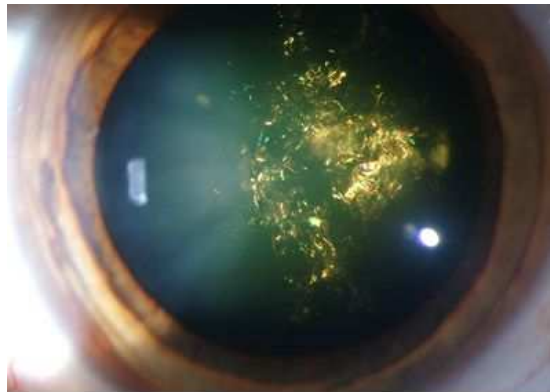


Figure 8. Eye affected by christmas tree/polychromatic cataract.

harder, longer, and riskier than when treated earlier on.

Finally **diabetic snowflake** is a rare type of cataract that can happen if you have diabetes. It gets worse quickly and forms a gray-white pattern that looks like a snowflake (figure 9).



Figure 9. Eye affected by diabetic snowflake cataract.

1.2.3. Surgery

Even if cataract surgery is one of the most common surgical procedures performed around the world and has a very high success rate, sometimes no medical treatment has been shown to be effective in the treatment or prevention of cataracts, although this is an active area of research. To slow the development of cataracts it is generally recommended that patients eat a balanced diet, prevent excessive exposure to UV radiation by using good quality UV blocking sunglasses, avoiding injuries by using protective eyewear, and if diabetic closely control blood sugar levels. Other approaches to temporarily improve visual function include careful refraction to get the best-corrected vision, pharmacological dilation, increased lighting and the use of magnifiers for near work.

In other cases cataract surgery is recommended and sometimes necessary. For example, doctors may recommend cataract surgery if a cataract makes it

difficult to examine the back of the eye, making hard to monitor or treat other eye problems, such as age-related macular degeneration or diabetic retinopathy. Besides, if cataract starts to affect vision, compromising the patient level of independence, then cataract surgery is necessary to restore the normal situation.

The most common type of cataract surgery in the United States utilizes ultrasound energy to break the cataract into particles small enough to aspirate through a handpiece. This technique is referred to as phacoemulsification. Other techniques include manual extracapsular cataract extraction (ECCE) in which the entire nucleus of the cataract is removed from the eye in one piece after extracting it from the capsular bag. While ECCE traditionally involved a large incision that required multiple sutures, a newer technique known by many names (such as manual small incision cataract surgery or small incision ECCE) allows for manual extraction without the need for any sutures. The goal in modern cataract surgery is not only the removal of the cataract, but also the replacement of the cataract with an intraocular lens (IOL). The IOL is typically placed during the cataract surgery, and may be placed in the capsular bag as a posterior chamber lens (PCIOL), in the ciliary sulcus, as a sulcus lens, or in the anterior chamber anterior to the iris as an anterior chamber lens (ACIOL). There are multiple types of IOLs that may be used in modern cataract surgery, including monofocal, multifocal, accommodative, and astigmatism-correcting lenses. The goal of all IOLs is to improve vision and limit dependency upon spectacles or contact lenses. Recently, the femtosecond laser, familiar to the refractive ophthalmologist for its role in LASIK, INTACS, and corneal transplantation, has been adapted to assist in cataract surgery. This procedure still relies upon the cataract surgeon to remove lens material in a manner similar to phacoemulsification, but it replaces several manual steps of the procedure with a more automated laser mechanism.

1.2.4. Diagnosis

To diagnose cataract several tests can be conducted, including:

- **Visual acuity test.** A visual acuity test uses an eye chart to measure how well a person can read a series of letters. The eyes are tested one at a time, while the other eye is covered. Using a chart or a viewing device with progressively smaller letters, the eye doctor determines if the subject has 20/20 vision or if the vision shows signs of impairment.
- **Slit-lamp examination.** A slit lamp allows the eye doctor to see the structures at the front of the eye under magnification. The microscope is called a slit lamp because it uses an intense line of light, a slit, to illuminate cornea, iris, lens, and the space between iris and cornea. The slit allows the doctor to view these structures in small sections, which makes it easier to detect any tiny abnormalities.
- **Retinal exam.** To prepare for a retinal exam, the eye doctor usually puts drops in the eyes to open pupils wide (dilate). This makes it easier to examine the back of the eyes (retina). Using a slit lamp or a special device called ophthalmoscope, the eye doctor can examine the lens for

signs of a cataract. These devices will be described more in details in the following chapter.

2. Diagnostic tools for ophthalmology

In 1999, the World Health Organization (WHO) launched an initiative called Vision 2020: The right to sight [7]. The objective of the initiative is to eliminate avoidable causes of blindness around the world and prevent the projected increase of avoidable visual impairment cases worldwide. Since then, more than 90 nongovernmental organizations, agencies, institutions, and corporations have pledged their support of this initiative. If successful, this would reduce the cases of blindness from 76 million to below 25 million. The program is based on several core principles: human resource development, infrastructure and technology development, disease control, advocacy, and collaboration among stakeholders in eye health [8].

Lot of different diseases can affect eye and/or visual system, form the most common and well-studied ones, such as glaucoma or macular degeneration to the ones rare or even still unknown, for example Bietti's Crystalline Dystrophy, a rare autosomal recessive ocular disease that involves yellow-white crystalline lipid deposits in the retina and sometimes cornea.

All these pathologies affect specific anatomical parts of the visual system and for this reason physicians have to use different tools or tests in order to perform a correct diagnosis. It is not difficult to imagine how wide could be the area regarding ophthalmological instrumentation. Since the main topics of this thesis are cataract and fundus images, the following paragraphs will present diagnostic tools regarding these subjects.

2.1. Slit lamp

Today the slit lamp (figure 10) is the ophthalmologist's most frequently used and most universally applicable examination instrument. The most important field of application is the examination of the anterior segment of the eye including the crystalline lens and the anterior vitreous body. Supplementary optics such as contact lenses and additional lenses permit observation of the posterior segments and the iridocorneal angle that are not visible in the direct optical path. A number of accessories have been developed for slit lamps extending their range of application from pure observation to measurement, such as for measuring the intraocular pressure [9].

The standard slit lamp is comprised of three elements:

- **Slit illumination system.** It is intended to produce a slit image that is as bright as possible, at a defined distance from the instrument with its length, width, and position being variable.
- **Stereomicroscope.** Its aim is to provide optimum stereoscopic observation with selectable magnification. The size of the field of view and the depth of field are expected to be as large as possible, and there should be enough space in front of the microscope for manipulation on the eye.
- **Mechanical system.** It connects the microscope to the illumination system and allowing for positioning of the instrument.



Figure 10. Example of a modern slit lamp.

The slit lamp is a multi-purpose instrument and it enables the user to inspect individual eye segments in quick succession to obtain a general impression of the eye and make a diagnosis. There are several methods of examination that can be performed with a slit lamp. It is possible to refer each of them to a specific illumination technique. In particular *direct focal illumination*, which is the most important type of illumination of this instrument, is the usual technique used to detect cataract.

2.1.1. Direct focal illumination

Direct focal illumination, also know as observation with an optical section, is the most frequently applied method of examination with the slit lamp. With this method, the axes of illuminating and viewing path intersect in the area of the anterior eye media to be examined (figure 11).

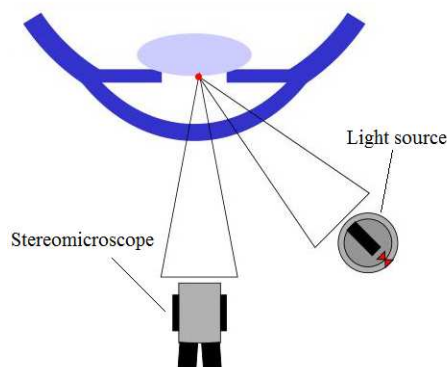


Figure 11. Working principle of a slit lamp in direct focal illumination setting.

The angle between illuminating and viewing path should be as large as possible (up to 90°), whereas the slit length should be kept small to minimise dazzling

the patient. With a narrow slit (about 0.1 mm to 0.2 mm) and a sufficiently small angular aperture, the illuminating beam takes the form of two knife blades placed edge to edge as shown in figure 12. Scattered light appears only in this "optical section". The intensity of scattered light depends on the object structures and increases with increasing slit brightness and the higher proportion of short-wave light obtained by an increased colour temperature of the light source.

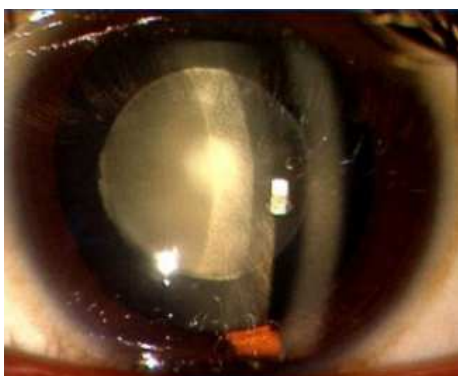


Figure 12. Slit lamp examination. Lens under direct focal illumination.

In conjunction with the stereomicroscope an optical section allows very precise depth information providing precise data of the shape of interfaces of transparent media. With a narrow slit and clear media, the images of slit and object appear sharply focused at the same time. Slit width and magnification may be varied depending on the object to be examined. With this method, brilliant optical section images can be obtained from the cornea through to the rear face of the crystalline lens.

With a narrow slit, the depth and position of different objects (e.g. the penetration depth of foreign bodies, shape of the lens etc.) can be resolved more easily. With a wide slit their extension and shape are visible more clearly (e.g. depth extension of injuries). It is therefore useful to vary the slit width during the examination.

The crystalline lens is particularly suited for viewing via an optical section where the discontinuity zones can be made visible with a narrow slit. For examination of the anterior segments of the vitreous body it is advisable to use the smallest possible slit length to avoid dazzling of both patient and examiner. In these examinations, slit brightness should be high.

The main applications of this technique can be summarized in the following items [9]:

- Illumination methods for features that stood out in diffuse illumination but could not be observed in detail; particularly suitable for the assessment of scars, nerves, vessels, etc.
- Observation by optical section is also of great importance for the determination of the stabilisation axis of toric contact lenses (typically used to correct astigmatism).
- Optical sections through the crystalline lens are also particularly good. Capsule, cortex, lens star and *cataracts* can be observed without difficulty.

2.2. Ophthalmoscope

The ophthalmoscope is considered to have been invented in 1851 by the German physiologist Hermann von Helmholtz, though it is sometimes credited to English mathematician and inventor Charles Babbage, who in 1847 developed an instrument thought to resemble the ophthalmoscope. This important instrument became a model for later forms of endoscopy. The device consists of a strong light that can be directed into the eye by a small mirror or prism. The light reflects off the retina and back through a small hole in the ophthalmoscope, through which the examiner sees a nonstereoscopic magnified image of the structures at the back of the eye, including the optic disk, retina, retinal blood vessels, macula, and choroid. The ophthalmoscope is particularly useful as a screening tool for various ocular diseases such as glaucoma or diabetic retinopathy. As said before, the light emitted by the ophthalmoscope is very bright and this forces the pupils to constrict and let pass through them insufficient light to perform the examination. To overcome this issue, physicians use some mydiatric in order to dilate the pupils.

2.2.1. Direct ophthalmoscope

For many decades the only typology of this device, the direct ophthalmoscope (figure 13) is a critical handheld tool used to inspect the back portion of the interior eyeball, the fundus oculi. The doctor holds the lens of this instrument over the pupil of the patient and looks through it by coming very close to the face of the patient. Examination is best carried out in a darkened room. The examiner looks for changes in the color or pigment of the fundus, changes in the caliber and shape of retinal blood vessels, and any abnormalities in the macula lutea.



Figure 13. Direct ophthalmoscope.

The direct ophthalmoscope emits a diverging beam of light into the eye of the patient, which illuminates the retina, reflecting light back towards the observer. An erect, virtual image of the retina is seen, which, dependent on

the refractive state of both observer and patient, may require focusing—either by accommodation or using the built-in focusing lenses of the ophthalmoscope.

The field of view and magnification are highly dependent on the refractive state of the patient, with myopes providing a greater degree of magnification and smaller field of view than emmetropic patients. The field of view is greater in patients with widely dilated pupils and also increases with increasing proximity to the patient.

The anterior segment of the eye can also be examined with the direct ophthalmoscope by using the built-in lenses and light as a self-illuminating loupe (magnifying glass). By increasing the lens power to $\approx +15$ D and observing the patient from ≈ 5 cm distance, a magnified view of these structures can be appreciated.

As the images involved are virtual, the optics of the situation mean that patient movement is grossly amplified. This means that small amounts of nystagmus (uncontrolled movements of the eye) can be easily picked up; however, with larger and more rapid excursions, indirect ophthalmoscopy may provide a better way of imaging the fundus [10].

Another reason to prefer indirect ophthalmoscope is the wider portion of retina which allows to inspect with. Indeed, even with appropriate illumination, direct ophthalmoscopy has a small field of view. Figure 14 shows that of four points in the fundus, points one and four cannot be seen because pencils of light emanating from these points diverge beyond the pupil of the observer.

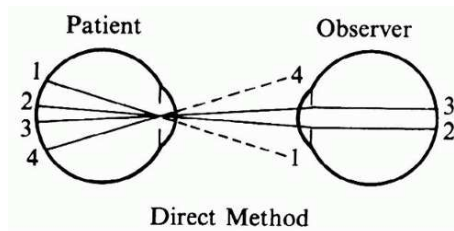


Figure 14. Limited field of view in the direct method. Peripheral pencils of light do not reach the pupil of the observer.

2.2.2. Indirect ophthalmoscope

The indirect ophthalmoscope (figure 15) is a more recent invention and constitutes a light placed on a headband worn by the physician. In addition to the light, a handheld lens helps in the examination of the retina and the fundus of the eye. Comparing direct vs indirect ophthalmoscope, the indirect ophthalmoscope delivers a stronger source of light, greater opportunity for stereoscopic inspection of the eyeball interior, and a specifically designed objective lens.

Even if the two instruments are used to achieve similar goals, indirect ophthalmoscopes have proven to be an exceptionally valuable device for the treatment and diagnosis of detachments, holes, and retinal tears. In order for the satisfactory use of an indirect ophthalmoscope, the patient's pupils must be completely dilated.



Figure 15. Indirect ophthalmoscope.

The principle of indirect ophthalmoscopy is to make the eye highly myopic by placing a strong convex lens in front of the eye of the patient so that the emergent rays from an area of the fundus are brought to focus as a real inverted image between the lens and the eye of the observer.

The use of the intermediate lens has several important implications that make indirect ophthalmoscopy more complicated than direct ophthalmoscopy. The primary purpose of the ophthalmoscopy lens is to bend pencils of light toward the pupil of the observer. Figure 16 also demonstrates one of the most characteristic side effects of this arrangement: compared with the image in direct ophthalmoscopy, the orientation of the image on the retina of the observer is inverted. For the novice, this often causes confusion in localization and orientation. Figure 16 further shows that in this arrangement the pupil of the patient is imaged in the pupillary plane of the observer. In optical terms the pupils are in conjugate planes.

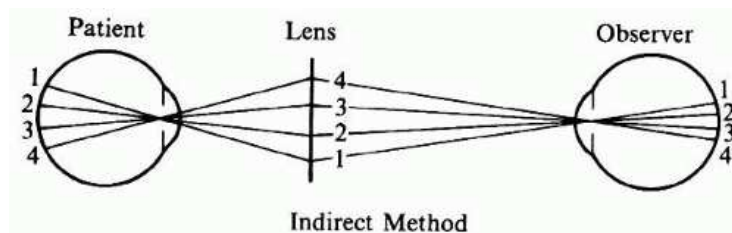


Figure 16. Extended field of view in the indirect method. The ophthalmoscopy lens redirects peripheral pencils of light toward the observer.

Indirect ophthalmoscopes can be divided into two different categories that include:

- **Monocular indirect ophthalmoscopes.** They offer a wider field of view and higher magnification levels than the traditional ophthalmoscope. As the name suggests, however, the monocular indirect ophthalmoscope only offers a single view of the interior of the eye. For a physician to

properly assess a patient’s ocular condition and fundus, you should have the patient look in multiple directions.

- **Binocular indirect ophthalmoscopes.** Instead of only projecting one, binocular indirect ophthalmoscopes project three elements in the eye. As a result, the ophthalmologist, or optometrist can get a three-dimensional rendition of the patient’s interior eye, which facilitates a much more thorough examination.

Table 1 summarizes all the principal differences between direct and indirect ophthalmoscope.

	Indirect ophthalmoscope	Direct ophthalmoscope
Observation view field diameter	Wide view ($\approx 37^\circ$ in diameter)	Small view ($\approx 10^\circ$ in diameter)
Magnification	5 times	15 times
Structures viewed	Peripheral retina seen	Central retina only
Brightness	More brightness	Less brightness
Stereopsis	Binocular indirect provides superior stereopsis	Image created isn’t stereoscopic
Image of fundus	Inverted and virtual image	Erect and real image
Visibility in hazy media	Decent	Poor

Table 1. Comparison between direct and indirect ophthalmoscope.

2.3. Fundus camera

The first photographs of the retina were published by Jackman and Webster in 1886. The next breakthrough was the first commercially available fundus camera produced by Carl Zeiss in 1926, following which considerable improvements to the field of view (FoV) were made. Through the years, camera systems have evolved to boast sharper images, nonmydriatic wide-field options, pupil tracking, and, most recently, portability. Popular manufacturers in the market today are Topcon, Zeiss, Canon, Nidek, Kowa, CSO, and CenterVue.

A fundus camera (figure 17) or retinal camera is a specialized low power microscope with an attached camera designed to photograph the interior surface of the eye, including the retina, retinal vasculature, optic disc, macula, and posterior pole (i.e. the fundus). The design of the traditional fundus camera system is based on monocular indirect ophthalmoscopy. The reference layout consists of a sequence of optic components including objective and condensing lenses, beam splitters, mirrors, masks, diffusers, and polarizers, which altogether direct the illuminating light through the pupil of the eye, collecting light reflected from the retinal surface and relaying it to imaging optics forming an image of the

retina on the detector screen. Advanced versions of these systems are equipped with additional features like automated analysis and algorithms. Filters can be applied to camera systems for autofluorescence, fundus fluorescein angiography, and indocyanine green angiography [11].



Figure 17. Fundus camera.

Retinal imaging presents a challenging difficulty considering that the retina must be illuminated and imaged simultaneously. For this reason most of fundus cameras are designed with internal structure, where the imaging path and illumination path share the common eyepiece and are combined by a beam splitter. Compared with the external structure, though the former has increased the degree of difficulty of the system design, the illumination is more efficient, and fewer lenses are needed. However, controlling the amount of light incident on the fundus within the security exposure dosage range is necessary. Taking into account the human eye's comfort under a wide angle of visible light, uniform illumination is also required when designing the illumination path [12].

The need for a miniature fundus camera device has emerged from specific limitations that accompany the use of traditional table-top fundus cameras. First, they form a bulky system, incorporating a host of optical and mechanical components, and the alignment of every part with respect to another is a critical parameter for good-quality images. Second, the operation of such a sophisticated system requires skilled personnel. Third, the bulkiness and complexity of the instrument restrict its use only in high-end clinical settings, such that it is difficult to be accessible in remote rural settings. Fourth, the number of optical components and add-on features in more recent devices renders the cost of the cameras exorbitantly high for them to be installed in rural locales where much of the population is subjected to ailments amounting to visual morbidity.

To overcome these issues lot of interesting novel technologies have been proposed in literature. One of the most promising innovation is the use of smartphones such as professional medical devices.

Smartphones equipped with faster processors, larger storage memory, smaller

batteries, and advanced operating systems have paved the way for numerous applications. Smartphones have become an integral part of the medical field lately to provide fast and clear access to electronically mailed digital images, instant messaging and virtual private network, user-interface services, and mobile healthcare computing devices. Research on the use of smartphones for medicine is growing. This technological advancements allow smartphone-based attachments and integrated lens adaptors to transform the smartphone into a fundus camera and others ophthalmological tools, revolutionizing modern ophthalmology [11].

2.4. D-EYE

D-EYE Srl is a leading developer of retinal screening systems for smartphones. The company is an Italian startup founded in 2014 with the mission of improving access to vital health services. To do that D-EYE Srl designs and manufactures diagnostic instruments, along with companion applications, that enable mass health screenings and data collection.

The flagship of the company is the homonymous device D-EYE (figure 18), a smartphone-based retinal imaging system. The retinal examiner is a phone-case-sized add-on that turns an Apple (Apple Inc., Cupertino, USA) smartphone into a fundus camera capable of taking high-definition images and video of the eye for health screening and evaluation. Specifically, the device consists in a magnetic fundus lens that attaches to an iPhone. It utilizes a user-friendly smartphone application and the built-in iPhone camera to take fundus photographs and videos. D-EYE is a FDA approved medical device, and its corresponding smartphone application is HIPAA compliant. This makes it an effective and simple method to capture, document, and consult within a single interface.



Figure 18. D-EYE device.

The design of the device is suitable both for humans and animals. Ophthalmologists can utilize D-EYE to diagnose several pathologies such as glaucoma, diabetic retinopathy, cataract, papilledema, optic glioma, macular hole, vitreous detachment, drusen, astrocytic hamartoma, optic disc dyskinesia. With the same instrument vets can perform retinal exams in dogs, cats, rabbits and other small animals, as well as horses, falcons and more.

All the captured photos are then used in documentation, follow-up, and discussion of complex cases with colleagues and patients alike. The D-EYE's simplistic, yet effective design, along with the image capturing and sharing capabilities of the iPhone, extends the system's potential utility to medical personnel with little ophthalmologic experience. This is a huge advantage in undeveloped areas of the world, where most patients do not have access to facilities with expensive conventional ophthalmoscopes. In these cases, when a whole team of clinical staff could not logistically travel to developing countries or no eye specialists are available, training local staff and remotely analyzing images could be a solution in helping to overcome eye care disparities abroad. The relatively low cost gives these areas the opportunity to have diagnosis made by qualified experts via teleophthalmology. More in general, telemedicine is probably the way how future will face the access of health service, especially in the developing world. Devices suitable for telemedicine, such as D-EYE, will be essential to archive this goal [13].



Figure 19. Retinal exam performed through smartphone-based ophthalmoscope in undeveloped areas.

D-EYE, as a smartphone-based device, is ideal for children (figure 20). A typical problem pediatricians have to deal with is that today's children might be afraid of a big ophthalmic instrument while touching and playing with a smartphone can make them feel more comfortable. Moreover also school environment can take advantages from this device. Professors can easily share real-case and also live photos or videos in order to offer a more effective way of teaching. On the other hand this low-cost, user-friendly device is perfect to be handled by students for learning and improving diagnostic skills.



Figure 20. D-EYE used for a children ophthalmological examination.

3. Deep Learning

During the last few decades Artificial Intelligence (AI) is getting more and more relevance in a very wide range of areas: e-commerce, medicine, automation, electronics, food companies and so on. Its developments are intimately linked to those of computing that have led computers to perform increasingly complex tasks, which could previously only be delegated to a human. Despite this, it is worth remembering that the origins of AI can be found long time ago. During IV sec. a.C. the Greek philosopher Aristotle spoke about logical deductive reasoning using syllogism. This concept can be found in every philosophy book: a conclusion is drawn from two given or assumed propositions (premises). Probably Aristotle was not thinking about AI while he was studying these contents, but this can be seen as a try to automate human thought.

One of the most important area of AI is machine-learning. Machine-learning systems are used to identify objects in images, transcribe speech into text, match news items, posts or products with users'interests, and select relevant results of search. Conventional machine-learning techniques were limited in their ability to process natural data in their raw form. For decades, constructing a pattern-recognition or machine-learning system required careful engineering and considerable domain expertise to design a feature extractor that transformed the raw data (such as the pixel values of an image) into a suitable internal representation or feature vector from which the learning subsystem, often a classifier, could detect or classify patterns in the input [14].

A peculiar edge of machine-learning is the so called deep learning. The power of deep learning is that it does not need any feature extraction passage so it can learn directly from raw data. In the early days of artificial intelligence, the field rapidly tackled and solved problems that are intellectually difficult for human beings, but relatively straight forward for computers, problems that can be described by a list of formal, mathematical rules. The real challenge for artificial intelligence is to solve tasks that are easy for people to perform but hard for people to formally describe. These tasks are intuitive for people, like recognizing spoken words or faces in images.

The solution is to allow computers to learn from experience and understand the world in terms of a hierarchy of concepts, with each concept defined through its relation to simpler concepts. By gathering knowledge from experience, this approach avoids the need for human operators to formally specify all the knowledge that the computer needs. The hierarchy of concepts enables computers to learn complicated concepts by building them out of simpler ones. Drawing a graph which shows how these concepts are built on top of each other, this graph is deep, with many layers. For this reason, this approach is called deep learning [15].

3.1. Artificial Neural Network (ANN)

Theoretical neurophysiology rests on certain cardinal assumptions. The nervous system is a net of neurons, each having a soma and an axon. Their adjunctions,

or synapses, are always between the axon of one neuron and the soma of another. At any instant a neuron has some threshold, which excitation must exceed to initiate an impulse [16]. As the fundamental unit of the nervous system is the neuron, also in every ANN there is a basic brick which is used to build the whole net. Usually it is called artificial neuron or perceptron and its function is to simulate the behaviour of a biological neuron. Figure 21 presents the principle of working of a perceptron. Basically there are many inputs (x_1, x_2, \dots, x_n) and each of them has his own weight (w_1, w_2, \dots, w_n). Weights can be either positive or negative, similar to the bias (b) a constant term. All these factors pass through the function $f = \sum_{i=1}^n x_i w_i + b$ and the result passes to the activation function. In the particular case of figure 21, the activation function is a step function, i.e. it produces 0 if s is negative or equal to 0 and 1 otherwise.

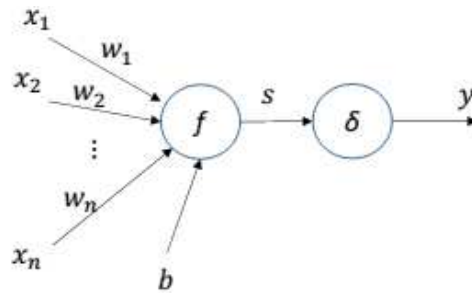
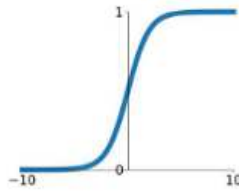


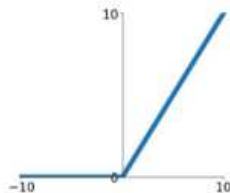
Figure 21. Basic scheme of how an artificial neuron works.

Step function δ is one of the simplest activation function. There are lot of different ones that can be used, here are some of the most important.

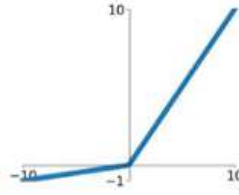
- **Sigmoid:** $y = \frac{1}{1 + e^{-s}}$



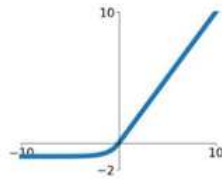
- **ReLU (Rectified Linear Unit):** $y = \max(0, s)$



- **Leaky ReLU:** $y = \max(\alpha s, s)$ (typically $\alpha=0.1$)



- **ELU:** $y = \begin{cases} \alpha(e^s - 1) & \text{if } s < 0 \\ s & \text{if } s \geq 0 \end{cases}$



Each activation function has his own peculiarity and can be more useful in some particular nets than in others, in according to their final purpose. Here the similarity between the biological world and the computer science one is evident. When the summation of some electrical stimuli exceeds a certain threshold the biological neuron fires an impulse. On the other hand there is also a threshold, designed with a mathematical function, that says when an artificial neuron has to fire.

As said before, artificial neurons are the fundamental unit of every ANN but the strength of the net rests in his connections. Each connection is characterized by the strength with which pairs of nodes are excited or inhibited. Positive values indicate excitatory connections, the negative ones inhibitory connections. The connections between the nodes can modify themselves over time. This dynamic starts a learning process in the entire ANN. The way through which the nodes modify themselves is called "Law of Learning". The total dynamic of an ANN is tied to time. In fact, for the ANN to modify its own connections, the environment has to necessarily act on the ANN more times. Data are the environment that acts on the ANN. The learning process is one of the key mechanisms that characterize the ANN, which are considered adaptive processing systems. The learning process is a way to adapt the connections of an ANN to the data structure that make up the environment and, therefore, a way to "understand" the environment and the relations that characterize it.

Neurons can be organized in any topological manner (e.g. one- or two-dimensional layers, three-dimensional blocks or more-dimensional structures), depending on the quality and amount of input data. The most common ANNs are composed as following [17]:

- A certain number of neurons is combined to an input layer, normally depending on the amount of input variables.
- The information is forwarded to one or more hidden layers working within the ANN.

- The output layer, as the last element of this structure, provides the result.

Figure 22 shows an example of the structure just mentioned.

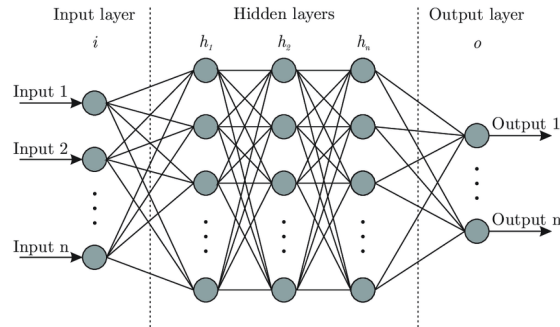


Figure 22. Artificial Neural Network structure.

There is not a single purpose for what an ANN is built up. These networks can be useful for quite different reasons and in order to understand these different functionalities it is important to understand how a network learns from input and how it can produce results.

Basically, there are two different ways of learning.

Supervised learning. The majority of practical machine learning uses supervised learning. Supervised learning is where there are input variables (x) and an output variable (Y) the algorithm is used to learn the mapping function from the input to the output.

$$Y = f(X)$$

The goal is to approximate the mapping function so well that when the algorithm analyses new input data (x), it can predict the output variables (Y) for that data. In supervised learning, the training phase of the machine is performed using data which is well "labelled". It means that some data are already tagged with the correct answer. It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process: the algorithm iteratively makes predictions on the training data and is corrected by the teacher. Learning stops when the algorithm achieves an acceptable level of performance. Supervised learning problems can be further grouped into regression and classification problems.

- **Classification:** A classification problem is when the output variable is a category, such as "red" or "blue" or "disease" and "no disease".
- **Regression:** A regression problem is when the output variable is a real value, such as "dollars" or "weight".

Unsupervised learning. Unsupervised learning is where there is only input data (X) and no corresponding output variables. For this reason it is not necessary supervise the model. Instead, it is required to allow the model to

work on its own to discover information. It mainly deals with the unlabelled data. The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data. These are called unsupervised learning because unlike supervised learning above there is no correct answers and there is no teacher. This process allows to perform more complex processing tasks compared to supervised learning. Although, unsupervised learning can be more unpredictable and gives result that has to be further confirmed.

Unsupervised learning problems can be further grouped into clustering and association problems.

- Clustering: A clustering problem tries to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.
- Association: An association rule learning problem tries to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

Since the work of this thesis is based on supervised learning network for image classification, the following part will focus on a more detailed explanation of this part.

In supervised learning, the learner (typically, a computer program) is learning provided with two sets of data, a *training set* and a *test set*. The training set consists of n ordered pairs $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, where each x_i is some measurement or set of measurements of a single example data point, and y_i is the label for that data point. The test data in supervised learning is another set of m measurements without labels: $(x_{n+1}, x_{n+2}, \dots, x_{n+m})$ [18]. The aim of the net is to label the test data point from some information obtained from the training set. In order to do that the training phase consists in a series of cyclical passages that bring to minimize a, so called, cost function.

A cost function, sometimes referred to as a loss or objective function, is somehow a mathematical representation of the distance between the target and the prediction of the net. The more this distance is short, the more accurate the prediction will be. An example of a simple but quite effective cost function is $C(w, b)$, the mean squared error or just MSE.

$$C(w, b) = \frac{1}{2n} \sum_x \|y(x) - a\|^2$$

Here, w denotes the collection of all weights in the network, b all the biases, n is the total number of training inputs, a is the vector of outputs from the network when x is input, and the sum is over all training inputs, x . Of course, the output a depends on x , w and b , but to keep the notation simple it is not explicitly indicated this dependence. The notation $\|v\|$ just denotes the usual length function for a vector v . Inspecting the form of the quadratic cost function, it is possible to notice that $C(w, b)$ is non-negative, since every term in the sum is non-negative. Furthermore, the cost $C(w, b)$ becomes small, i.e., $C(w, b) \approx 0$, precisely when $y(x)$ is approximately equal to the output, a , for all training inputs, x . So, the training algorithm has done a good job if it can find weights and biases so that $C(w, b) \approx 0$. By contrast, it is not doing so well when $C(w, b)$ is large, that would mean that $y(x)$ is not close to the output a for a large number

of inputs. So the aim of a training algorithm is to minimize the cost $C(w, b)$ as a function of the weights and biases. In other words, the training aims to find a set of weights and biases which make the cost as small as possible. This can be done using an algorithm known as gradient descent [19].

This kind of algorithm calculates iteratively the derivative of the cost function with respect to the weights in order to properly update them. Thanks to the derivative it is possible to know locally where the function is going and since the aim is to minimize this function, the algorithm will set the parameters for the following iteration aiming to have a lower cost value. The procedure just mentioned is repeated several times until a minimum (not always the global one) is reached. Figure 23 presents an ideal representation of this approach. It is important to highlight that, in real cases, cost functions are not as simple as a parabola, they are n -variable function where n is the number of weight and usually have more than one minimum. Moreover it is important to mention some other issues linked to gradient descent such as the choice of learning step and the vanishing gradient descent. Since the purpose of this chapter is to make a brief deep learning introduction and these topics need a deeper investigation, they are not explained in details in this thesis.

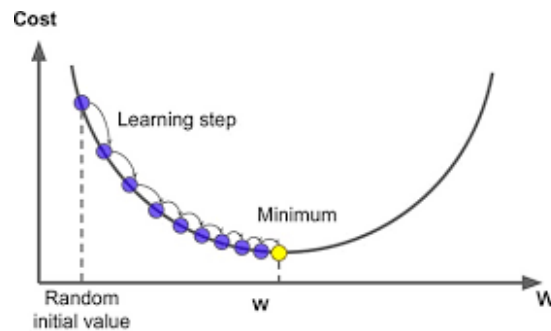


Figure 23. Exemplification of gradient descent algorithm.

After the description of supervised learning, it is worth mentioning some more details about what is the process of classification and present few examples.

In this type of task, the computer program is asked to specify which of k categories an input belongs to. To solve this, the learning algorithm is usually asked to produce a function $f : \mathbb{R}^n \rightarrow \{1, \dots, k\}$. When $y = f(x)$, the model assigns an input described by vector x to a category identified by numeric code y . There are other variants of the classification task, for example, where f outputs a probability distribution over classes.

An example of a classification task is object recognition, where the input is an image (usually described as a set of pixel brightness values), and the output is a numeric code identifying the object in the image. For example, the Willow Garage PR2 robot is able to act as a waiter that can recognize different kinds of drinks and deliver them to people on command [20]. Modern object recognition is best accomplished with deep learning. Object recognition is the same basic technology that enables computers to recognize faces, which can be used to automatically tag people in photo collections and for computers to interact more naturally with their users [15].

There are different types of artificial neural network that can be used for

classification task. One in particular is widely utilized and will be also the core of the algorithm presented in this work for retinal images classification. The following paragraph will be focused on *convolutional neural networks*.

3.2. Convolutional Neural Network (CNN)

Convolutional neural networks, or CNNs, are a specialized kind of neural network for processing data that has a known grid-like topology. Examples include time-series data, which can be thought of as a 1-D grid taking samples at regular time intervals, and image data, which can be thought of as a 2-D grid of pixels. Convolutional networks have been tremendously successful in practical applications. The name “convolutional neural network” indicates that the network employs a mathematical operation called convolution. Convolution is a specialized kind of linear operation, that CNNs use in place of general matrix multiplication in at least one of their layers [15].

Convolution is a simple mathematical operation which is fundamental to many common image processing operators. This operation provides a way of ‘multiplying together’ two arrays of numbers, generally of different sizes, but of the same dimensionality, to produce a third array of numbers of the same dimensionality. This can be used in image processing to implement operators whose output pixel values are linear combinations of certain input pixel values.

In an image processing context, one of the input arrays is normally just a graylevel image. The second array is usually much smaller, and is also two-dimensional (although it may be just a single pixel thick), and is known as the *kernel*. Figure 24 shows an example image and kernel that is used to illustrate convolution.

I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	I_{16}	I_{17}	I_{18}	I_{19}
I_{21}	I_{22}	I_{23}	I_{24}	I_{25}	I_{26}	I_{27}	I_{28}	I_{29}
I_{31}	I_{32}	I_{33}	I_{34}	I_{35}	I_{36}	I_{37}	I_{38}	I_{39}
I_{41}	I_{42}	I_{43}	I_{44}	I_{45}	I_{46}	I_{47}	I_{48}	I_{49}
I_{51}	I_{52}	I_{53}	I_{54}	I_{55}	I_{56}	I_{57}	I_{58}	I_{59}
I_{61}	I_{62}	I_{63}	I_{64}	I_{65}	I_{66}	I_{67}	I_{68}	I_{69}

K_{11}	K_{12}	K_{13}
K_{21}	K_{22}	K_{23}

Figure 24. An example small image (left) and kernel (right) to illustrate convolution.

The convolution is performed by sliding the kernel over the image, generally starting at the top left corner, so as to move the kernel through all the positions where the kernel fits entirely within the boundaries of the image. (Note that implementations differ in what they do at the edges of images, as explained below). Each kernel position corresponds to a single output pixel, the value of which is calculated by multiplying together the kernel value and the underlying image pixel value for each of the cells in the kernel, and then adding all these numbers together.

So, in the example, the value of the first pixel (top left) in the output image (O) will be given by:

$$O_{11} = I_{11}K_{11} + I_{12}K_{12} + I_{13}K_{13} + I_{21}K_{21} + I_{22}K_{22} + I_{23}K_{23}$$

If the image has M rows and N columns, and the kernel has m rows and n columns, then, with this procedure, the size of the output image will have $M - m + 1$ rows, and $N - n + 1$ columns.

Mathematically the convolution can be written as:

$$O(i, j) = \sum_{k=1}^m \sum_{l=1}^n I(i+k-1, j+l-1)K(k, l)$$

where i runs from 1 to $M - m + 1$ and j runs from 1 to $N - n + 1$.

Note that many implementations of convolution produce a larger output image than this because they relax the constraint that the kernel can only be moved to positions where it fits entirely within the image. Instead, these implementations typically slide the kernel to all positions where just the top left corner of the kernel is within the image. Therefore the kernel "overlaps" the image on the bottom and right edges. One advantage of this approach is that the output image is the same size as the input image. Unfortunately, in order to calculate the output pixel values for the bottom and right edges of the image, it is necessary to add input pixel values for places where the kernel extends off the end of the image. This strategy is usually called *padding*. Typically pixel values of zero are chosen for regions outside the true image, but this can often distort the output image at these places. Therefore another common approach (when it is possible) is to clip the filtered image to remove these spurious regions. Removing $n - 1$ pixels from the right hand side and $m - 1$ pixels from the bottom will fix things.

Next paragraph explains why CNNs use convolution and not some of the various other operation for imagine processing.

Figure 25 shows a color input image (25a) and its convolution results using two different kernels (25b and 25c). A 3×3 convolution matrix $K =$

$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$ is used. The convolution kernel should be of size $3 \times 3 \times 3$, in

which the user set every channel to K . When there is a horizontal edge at location (x, y) (i.e., when the pixels at spatial location $(x + 1, y)$ and $(x - 1, y)$ differ by a large amount), the convolution result to have high magnitude. As shown in figure 25b, the convolution results indeed highlight the horizontal edges. When the user set every channel of the convolution kernel to K^T (the transpose of K), the convolution result amplifies vertical edges, as shown in figure 25c. The matrix (or filter) K and K^T are called the Sobel operators.

If a bias term is added to the convolution operation, it can make the convolution result positive at horizontal (vertical) edges in a certain direction (e.g., a horizontal edge with the pixels above it brighter than the pixels below it), and negative at other locations. If the next layer is a ReLU activation function layer (it will be described later), the output of the next layer defines many "edge detection features", which activate only at horizontal or vertical edges in certain directions. By combining horizontal and vertical gradients, the Sobel kernels can highlight edges with any angles.

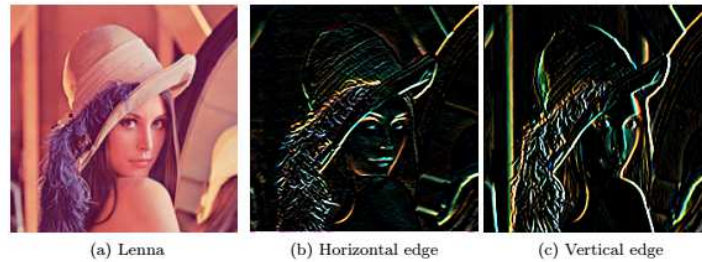


Figure 25. The Lenna image and the effect of different convolution kernels.

Moving further down in the deep network, subsequent layers can learn to activate only for specific (but more complex) patterns, e.g., groups of edges that form a particular shape. These more complex patterns will be further assembled by deeper layers to activate for semantically meaningful object parts or even a particular type of object, e.g., dog, cat, tree, beach, etc. One more benefit of the convolution layer is that all spatial locations share the same convolution kernel, which greatly reduces the number of parameters needed for a convolution layer. For example, if multiple dogs appear in an input image, the same "dog-head-like pattern" feature will be activated at multiple locations, corresponding to heads of different dogs. In a deep neural network setup, convolution also encourages parameter sharing. For example, suppose "dog-head-like pattern" and "cat-head-like pattern" are two features learned by a deep convolutional network. The CNN does not need to devote two sets of disjoint parameters (e.g., convolution kernels in multiple layers) for them. The CNN's bottom layers can learn "eye-like pattern" and "animal-fur-texture pattern", which are shared by both these more abstract features. In short, the combination of convolution kernels and deep and hierarchical structures are very effective in learning good representations (features) from images for visual recognition tasks.

It is important to add a note here. Although the paragraph uses phrases such as "dog-head-like pattern", the representation or feature learned by a CNN may not correspond exactly to semantic concepts such as "dog's head". A CNN feature may activate frequently for dogs' heads and often be deactivated for other types of patterns. However, there are also possible false activations at other locations, and possible deactivations at dogs' heads. In fact, a key concept in CNN (or more generally deep learning) is distributed representation. For example, suppose the task is to recognize N different types of objects and a CNN extracts M features from any input image. It is most likely that any one of the M features is useful for recognizing all N object categories; and to recognize one object type requires the joint effort of all M features [21].

Convolution is the main part of a convolutional layer, the layer that characterizes a CNN. Lot of other layers are proposed in literature to complete the whole structure of a convolutional neural network. Now, after the description of the aforesaid layer, the mostly common used ones will be presented.

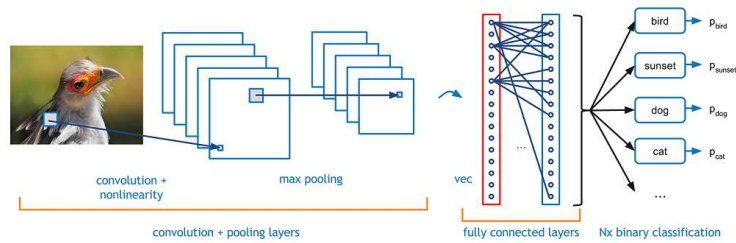


Figure 26. Example of CNN structure with explicit layers.

3.2.1. Convolutional layer

Last paragraph already described what concern convolution. Convolutional layers simply apply this mathematical operator to the input. As explained before, convolutional layers are responsible for detecting certain local features in all locations of their input images. To detect local structures, each node in a convolutional layer is connected to only a small subset of spatially connected neurons in the input image channels. To enable the search for the same local feature throughout the input channels, the connection weights are shared between the nodes in the convolutional layers [22].

It is worth mentioning that every layer has his own hyperparameter and a good tuning of them is essential to archive great performances in whatever kind of neural network.

In this particular layer there are three different hyperparameter: depth, stride and zero-padding.

- **Depth.** The depth of the output volume produced by the convolutional layers can be manually set through the number of neurons within the layer to a the same region of the input. This can be seen with other forms of ANNs, where neurons in the hidden layer are directly connected to every single neuron beforehand. Reducing this hyperparameter can significantly minimise the total number of neurons of the network, but it can also significantly reduce the pattern recognition capabilities of the model.
- **Stride.** It is also important to define the stride, that permits to set the depth around the spatial dimensionality of the input in order to place the receptive field. For example if the user set the stride to 1, then the algorithm will have a heavily overlapped receptive field producing extremely large activations. Alternatively, setting the stride to a greater number will reduce the amount of overlapping and produce an output of lower spatial dimensions. In other words is of how many pixel the filter is moving through the image for every step of convolution.
- **Zero-padding.** Zero-padding is the simple process of padding the border of the input, consist in adding fictitious pixel with 0 value and it is also an effective method to give further control as to the dimensionality of the output volumes.

It is important to note that these techniques alter the spatial dimensionality of the convolutional layers output. The following formula calculates it:

$$\frac{(V - R) + 2Z}{S + 1}$$

Where V represents the input volume size (height x width x depth), R represents the receptive field size, Z is the amount of zero padding set and S referring to the stride. If the calculated result from this equation is not equal to a whole integer then the stride has been incorrectly set, as the neurons will be unable to fit neatly across the given input [23].

3.2.2. ReLU layer

The rectified linear unit (commonly shortened to ReLU) aims to apply an "elementwise" activation function in order to keep activate some neurons while switch off some others. Even if activation functions were previously described, it is worth focusing on the ReLU one since it is one of the most utilized.

A ReLU layer does not change the size of the input. In fact, this layer can be regarded as a truncation performed individually for every element in the input. There is no parameter inside a ReLU layer, hence no need for parameter learning in this layer. As mentioned before the math behind this operation is simple and can be well described by the following formula:

$$y = \max(0, s) = \begin{cases} 0 & \text{if } s < 0 \\ s & \text{if } s \geq 0 \end{cases}$$

The purpose of ReLU is to increase the nonlinearity of the CNN. Since the semantic information in an image is obviously a highly nonlinear mapping of pixel values in the input, also the mapping from CNN input to its output should be highly nonlinear in order to be consistent. The introduction of ReLU to replace sigmoid is an important change in CNN, which significantly reduces the difficulty in learning CNN parameters, i.e. it reduces training time and also improves its accuracy.[21, 23]

3.2.3. Pooling layer

Pooling layers aim to gradually reduce the dimensionality of the representation, and thus further reduce the number of parameters and the computational complexity of the model. Moreover is it possible to identify the role of the pooling layer in merging semantically similar features into one. Because the relative positions of the features forming a motif can vary, reliably detecting the motif can be done by coarse-graining the position of each feature.

The pooling layer operates over each activation map in the input, and scales its dimensionality using the "MAX" function. In most CNNs, these come in the form of max-pooling layers with kernels of a dimensionality of 2 x 2 applied with a stride of 2 along the spatial dimensions of the input. This scales the activation map down to 25% of the original size whilst maintaining the depth volume to its standard size. *Stride* and *filter size* are the only two hyperparameter of this layer. To see a more detail about them refers to subparagraph 2.2.1.

Due to the destructive nature of the pooling layer, there are only two generally observed methods of max-pooling. The one just mentioned and furthermore overlapping pooling may be utilised, where the stride is set to 2 with a kernel size set to 3. Having a kernel size above 3 will usually greatly decrease the performance of the model.

It is also important to note that beyond max-pooling, CNN architectures may contain general-pooling. General pooling layers are comprised of pooling neurons that are able to perform a multitude of common operations including L1/L2-normalisation, and average pooling [14, 23].

3.2.4. Fully connected layer

A fully connected layer refers to a layer where the computation of any element in the output requires all elements in the input. A fully connected layer is sometimes useful at the end of a deep CNN model. For example, if after many convolution, ReLU and pooling layers, the output of the current layer contains distributed representations for the input image, the following step could be using all these features in the current layer in order to build features with stronger capabilities in the next one. A fully connected layer mainly used for this purpose. Moreover it is possible to implement this kind of layer starting with a convolutional layer with a kernel size equal to the size of the input matrix [21].

3.2.5. Output layer

The fully connected layer outputs a vector of K dimensions where K is the number of classes that the network will be able to predict. This vector contains the "probabilities" for each class of any image, or object in general, being classified. The final layer of the CNN architecture uses a *softmax* function to provide the classification output [24].

The Softmax regression is a form of logistic regression that normalizes an input value into a vector of values that follows a probability distribution whose total sums up to 1. This allows the output to be interpreted directly as a probability. Similarly, softmax functions are multi-class sigmoids, meaning they are used in determining probability of multiple classes at once.

The standard softmax function is defined by the formula:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \text{ for } i = 1, \dots, K \text{ and } z = (z_1, \dots, z_K)$$

Where K , as said before, is the number of classes and z a vector with all the nodes of the previous layer.

In multicategory classification, standard techniques typically treat all classes equally. This treatment can be problematic when the dataset is unbalanced in the sense that certain classes have very small class proportions compared to others. The minority classes may be ignored or discounted during the classification process due to their small proportions. This can be a serious problem if those minority classes are important. Talking about diagnosis in medical

field, when the minority classes represent the pathological situation, this kind of misclassification could be fatal.

Intuitively, one can put a relatively big weight for a minority class so that it cannot be ignored easily. For example, consider a binary problem with the class 1 to be the minority class and class 2 to be the majority class. Suppose the weights to be $(4, 1)$ for the two classes. This implies that one misclassified point of class 1 is treated to be equivalent to four misclassified points of class 2. Using a bigger weight for class 1, one can increase the impact of class 1 for the classification rule so that it will not be ignored due to its small proportion.

A natural choice of weights is to make use of the true proportions $\{\pi_j\}$ of different classes, if they are available. Let $(w_1 : w_2 : \dots : w_k) = (1/\pi_1 : 1/\pi_2 : \dots : 1/\pi_k)$. Using this choice, the aim is to put a big weight for class j if π_j is small. This indicates that this choice of weights can eliminate the unbalanced data effect because it is equivalent to finding a classifier that minimizes the mean within group error rate. Even if typically $\{\pi_j\}$ are not available there is an easy way to estimate them. A natural estimator is $\hat{\pi}_j = n_j/n$, where n_j represents the number of observations for class j in the training dataset (with size n). Using $\hat{\pi}_j$ could have some drawbacks. Clearly, the accuracy of the estimator $\hat{\pi}_j$ affects the performance. The smaller n_j and n are, the less reliable $\hat{\pi}_j$ is as an estimator of π_j . Since, in order to have good performances, with CNNs it is necessary to use huge dataset, this estimation can be considered reliable [25].

This kind of approach will be used even in the proposed algorithm and it will be further described in the following chapter.

3.3. Training phase

Training a CNN is the most critical part of a deep learning algorithm. The choice of the initial hyperparameters and the setting of the best training options can determinate either good or bad performances of the whole classification process.

To better understand the training phase it is necessary to first define what is the meaning of *batch* and *epoch*.

- The **batch size** is a hyperparameter that defines the number of samples to work through before updating the internal model parameters. Think of a batch as a for-loop iterating over one or more samples and making predictions. At the end of the batch, the predictions are compared to the expected output variables and an error is calculated. From this error, the update algorithm is used to improve the model, e.g. move down along the error gradient.
- The number of **epochs** is another hyperparameter that defines the number of times that the learning algorithm will work through the entire training dataset. One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters. An epoch is comprised of one or more batches. For example, an epoch that has one batch is called the batch gradient descent learning algorithm. Simplifying it can be thought as a for-loop over the number of epochs where each loop proceeds over the training dataset. Within this for-loop is another nested for-loop that iterates over each batch of samples, where one batch

has the specified "batch size" number of samples. The number of epochs is traditionally large, often hundreds or thousands, allowing the learning algorithm to run until the error from the model has been sufficiently minimized.

It is common to create line plots that show epochs along the x-axis as time and the error or skill of the model on the y-axis. These plots are sometimes called learning curves (figure 27). These plots can help to diagnose whether the model has over learned, under learned, or is suitably fit to the training dataset.

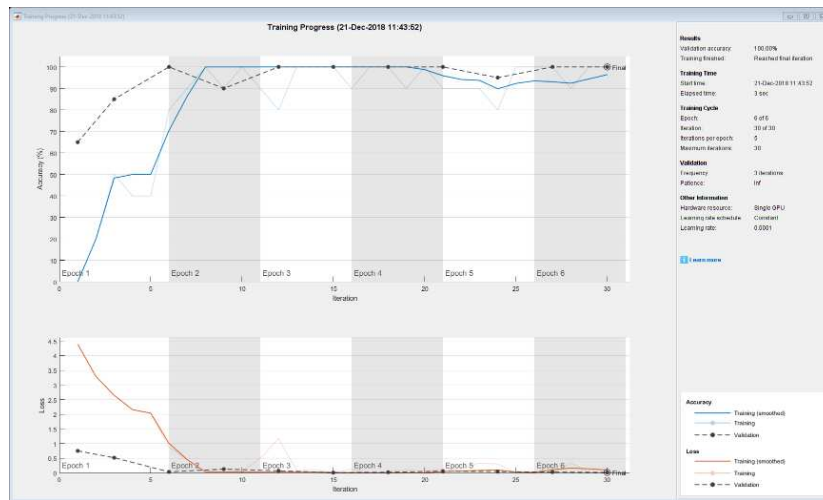


Figure 27. Training progress plot produced with MATLAB.

3.3.1. Overfitting and underfitting

Two very common issues every network can run into while training are *underfitting* and *overfitting*. A general definition of these concepts will be now presented [26]. Let M be a model and m be what has to be modelled.

- M is **overfitting** m if M does not generalize and is sensitive to particularities in m . In an extreme case, M could merely be a representation of m without any inference. A mining algorithm is producing overfitting models if the removal or addition of a small percentage of the process instances in m would lead to a remarkably different model. In a complex process with many possible paths, most process instances will follow a path not taken by other instances in the same period. Therefore, it is undesirable to construct a model that allows only for the paths that happened to be present in m as this is only a fraction of all possible paths. If one knows that only a fraction of the possible event sequences are in m , the only way to avoid overfitting is to generalize and have a model M that allows for more behavior than recorded in m .
- M is **underfitting** m if M allows for "too much behavior" that is not supported by m . This is also referred to as "overgeneralization". It is

very easy to construct a model that allows for the behavior seen in m but also completely different behavior. For example, assume a m consisting of 1,000 cases. For each case A is followed by B and there are no cases where B is followed by A. Obviously, one could derive a causal dependency between A and B. However, one could also create a model M where A and B are in parallel. The latter would not be "wrong" in the sense that the behavior seen in m is possible according to the model. However, it is very unlikely and therefore one could argue that M is underfitting m .

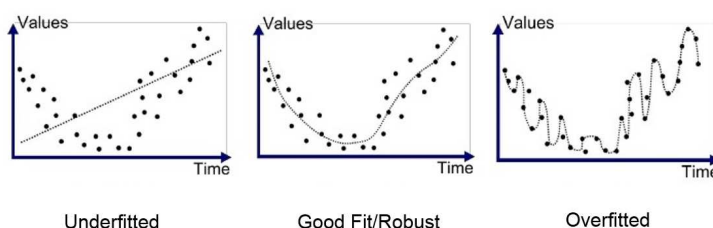


Figure 28. Comparison between underfitted, overfitted and a good-fit model.

Focusing on convolutional neural network, there is a way to understand if the model is underfitting or overfitting. It is revelatory to look at the accuracy of the *validation set* during the training. Typically the whole dataset is divided into: *training set*, *validation set* and *testing set*.

- **Training set.** The sample of data used to fit the model. The actual dataset that is used to train the model (weights and biases in the case of Neural Network). The model sees and learns from this data.
- **Validation set.** The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model, but this is for frequent evaluation. Hence the model occasionally sees this data, but never does it "learn" from this. For this reason it is important to follow the trend of the validation set accuracy. If the training accuracy increase while the validation one do not increase with the same velocity, decrease or remains stable there is a possible overfitting. This means, as shown in figure 28, that the model is too complex. Underfitting is characterized by the two curves of validation and training do not increase during the process. This is probably due to a too much simple model (figure 28).
- **Testing set.** The sample of data used to provide an unbiased evaluation of a final model fit on the training dataset. The Test dataset provides the gold standard used to evaluate the model. It is only used once a model is completely trained (using the train and validation sets). The test set is generally what is used to evaluate competing models in order to find the best one. Many a times the validation set is used as the test set, but it is not good practice. The test set should be well curated. It has to contain

carefully sampled data that spans the various classes that the model would face, when used in the real world.

The partition between these three parts of the dataset depends on the network target and the amount and the sample types. Despite this it is a good practise divide the initial dataset with about the following percentage: 70% training set, 30% testing set and 20% of the training set for the validation one.

3.3.2. Early stopping and drop out

When training a large network, there will be a point during training when the model will stop generalizing and start learning the statistical noise in the training dataset. This overfitting of the training dataset will result in an increase in generalization error, making the model less useful at making predictions on new data. The challenge is to train the network long enough that it is capable of learning the mapping from inputs to outputs, but not training the model so long that it overfits the training data. The idealized expectation is that during training the generalization error of the network evolves as shown in figure 29.

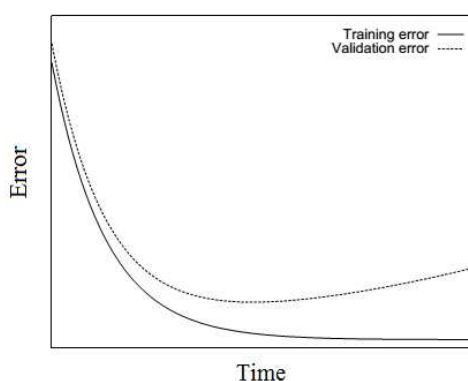


Figure 29. Idealized training and validation error curves.

Early stopping is widely used because it is simple to understand and implement and has been reported to be superior to regularization methods in many cases [27]. However in real cases it not so simple to find when is the right time to stop training and many different stopping criteria are proposed in literature. Another important reason why early stopping is used for regularization is because it does not interfere with backprop's ability to control capacity locally. Early stopping combined with backprop is so effective that very large nets can be trained without significant overfitting [28].

Dropout is a regularization method that approximates training a large number of neural networks with different architectures in parallel. During training, some number of layer outputs are randomly ignored or "dropped out". This has the effect of making the layer look-like and be treated-like a layer with a different number of nodes and connectivity to the prior layer. In effect, each update to

a layer during training is performed with a different "view" of the configured layer.

Dropout has the effect of making the training process noisy, forcing nodes within a layer to probabilistically take on more or less responsibility for the inputs. This conceptualization suggests that perhaps dropout breaks-up situations where network layers co-adapt to correct mistakes from prior layers, in turn making the model more robust.

Dropout simulates a sparse activation from a given layer, which interestingly, in turn, encourages the network to actually learn a sparse representation as a side-effect. As such, it may be used as an alternative to activity regularization for encouraging sparse representations in autoencoder models.

Because the outputs of a layer under dropout are randomly subsampled, it has the effect of reducing the capacity or thinning the network during training. As such, a wider network, e.g. more nodes, may be required when using dropout [29].

3.3.3. Data augmentation

Data augmentation is a strategy that enables practitioners to significantly increase the diversity of data available for training models, without actually collecting new data. Data augmentation techniques such as cropping, padding, and horizontal flipping are commonly used to train large neural networks.

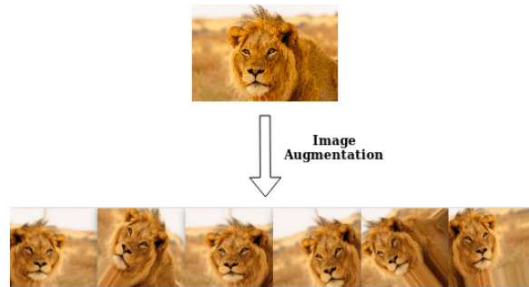


Figure 30. Example of data augmentation for image classification.

The motivation behind the use of data augmentation is both broad and specific. Specialized image and video classification tasks often have insufficient data. This is particularly true in the medical industry, where access to data is heavily protected due to privacy concerns. Important tasks such as classifying cancer types are hindered by this lack of data. Techniques have been developed which combine expert domain knowledge with pre-trained models. Similarly, small players in the AI industry often lack access to significant amounts of data.

Data augmentation has been shown to produce promising ways to increase the accuracy of classification tasks [30]. To increase the performances of the network, the algorithm implements this approach to improve the effectiveness of the training phase.

3.3.4. Transfer learning

Considering the context of deep learning most models which solve complex problems need a whole lot of data, and getting vast amounts of labeled data for supervised models can be really difficult, considering the time and effort it takes to label data points. A simple example would be the *ImageNet dataset* [31], which has millions of images pertaining to different categories, thanks to several years work starting at Stanford.

However, getting such a dataset for every domain is tough. Besides, most deep learning models are very specialized to a particular domain or even a specific task. While these might be state-of-the-art models, with really high accuracy and beating all benchmarks, it would be only on very specific datasets and end up suffering a significant loss in performance when used in a new task which might still be similar to the one it was trained on. This forms the motivation for transfer learning, which goes beyond specific tasks and domains, and tries to see how to leverage knowledge from pre-trained models and use it to solve new problems.

In order to give a strict definition of transfer learning, it is necessary to introduce what is a "domain" and a "task".

For this purpose, a *domain* \mathcal{D} consists of two components: a feature space \mathcal{X} and a marginal probability distribution $P(X)$, where $X = \{x_1, \dots, x_n\} \in \mathcal{X}$. For example, if the learning task is document classification, and each term is taken as a binary feature, then \mathcal{X} is the space of all term vectors, x_i is the i^{th} term vector corresponding to some documents, and X is a particular learning sample. In general, if two domains are different, then they may have different feature spaces or different marginal probability distributions.

Given a specific domain, $\mathcal{D} = \{\mathcal{X}, P(X)\}$, a *task* consists of two components: a label space \mathcal{Y} and an objective predictive function $f(\cdot)$ (denoted by $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$), which is not observed but can be learned from the training data, which consist of pairs $\{x_i, y_i\}$, where $x_i \in X$ and $y_i \in \mathcal{Y}$. The function $f(\cdot)$ can be used to predict the corresponding label, $f(x)$, of a new instance x . From a probabilistic viewpoint, $f(x)$ can be written as $P(y|x)$. In the document classification example, \mathcal{Y} is the set of all labels, which is True, False for a binary classification task, and y_i is "True" or "False".

For simplicity, in this survey, only the case where there is one source domain \mathcal{D}_S , and one target domain, \mathcal{D}_T is considered. This is by far the most popular of the research works in the literature. More specifically, the source domain data can be represented as $D_S = \{(x_{S_1}, y_{S_1}), \dots, (x_{S_{n_S}}, y_{S_{n_S}})\}$, where $x_{S_i} \in \mathcal{X}_S$ is the data instance and $y_{S_i} \in \mathcal{Y}_S$ is the corresponding class label. In this document classification example, D_S can be a set of term vectors together with their associated true or false class labels. Similarly, the target domain data representation is $D_T = \{(x_{T_1}, y_{T_1}), \dots, (x_{T_{n_T}}, y_{T_{n_T}})\}$, where the input x_{T_i} is in \mathcal{X}_T and $y_{T_i} \in \mathcal{Y}_T$ is the corresponding output. In most cases, $0 \leq n_T \ll n_S$.

Now it is possible to give a unified definition of transfer learning.

Given a source domain \mathcal{D}_S and learning task \mathcal{T}_S , a target domain \mathcal{D}_T and learning task \mathcal{T}_T , transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in \mathcal{D}_T using the knowledge in \mathcal{D}_S and \mathcal{T}_S , where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$ [32].

Once given a definition of transfer learning, it is important to note where the

network can take the "learning" from. In literature several pre-trained network were developed and are now available on-line. Some of the most famous are: AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan & Zisserman, 2014), GoogleNet (Szegedy et al., 2015), and ResNet (He et al., 2016) [33].

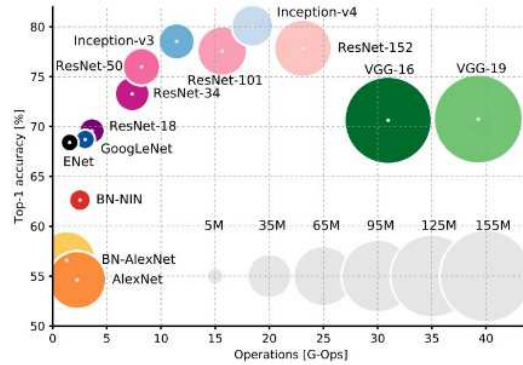


Figure 31. Comparison between different Deep Neural Networks.

Figure 31 shows the relation between accuracy, number of parameters (indicated with the size of the circles) and the amount of operations required for a single forward pass of different neural network architectures. The big amount of public neural network offers new researches an important opportunity to save time and to reach better performances. For this reason transfer learning is one of the first approach to solve image classification problem. Even if deeper analysis then suggest to built ad-hoc CNN, a better option for this very peculiar target, transfer learning gives quick initial results and so a way to follow.

3.4. Deep learning in Biomedical imaging

3.4.1. History of ML in computer vision and medical imaging fields

Before considering the last innovations deep learning brings in medical world, a brief description of the history of ML in the fields of computer vision and medical imaging is reported.

Until 1980, even when the term "machine learning" did not exist, classical classifiers such as *linear discriminant analysis* (LDA), *quadratic discriminant analysis* QDA, and *k-nearest neighbor* classifier (k-NN) were used for classification. In 1986, *multi-layer perceptron* (MLP) was proposed by Rumelhart and Hinton [34]. The MLP created the second neural network research boom (the first one was in 1960s). In 1995, Vapnik proposed *support vector machine* (SVM) [35] and became the most popular classifier for a while, partially because of publicly available code on the Internet. Various ML methods were proposed, including *random forests* by Ho et al. in 1995 [36], and *dictionary learning* by Mairal et al. in 2009 [37]. On the other hand, various ML with image input techniques were proposed before the introduction of the term "deep learning".

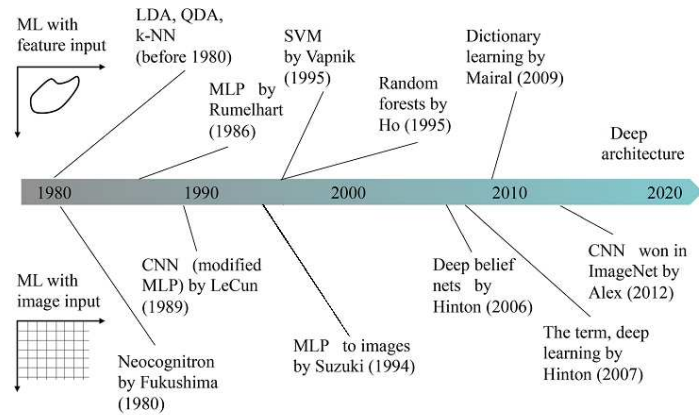


Figure 32. History of machine learning (ML) in the fields of computer vision and medical imaging. Upper part, feature-based ML. Bottom part, image-based ML.

It started from the *Neocognitron* by Fukushima in 1980 [38]. In 1989, LeCun et al. simplified the Neocognitron and proposed a CNN [39], but he did not study CNNs very much until recently. In 1994, Suzuki et al. applied an MLP to cardiac images in a convolutional way [40]. Some years later, in 2000, they proposed neural edge enhancers [41]. Hinton et al. proposed a *deep belief network* (DBN) in 2006 [42], and they created the term "deep learning" a year later. Deep learning was not recognized much until late 2012. In 2012, a CNN won in the ImageNet competition [43]. After that deep learning approach start getting more and more popularity in various fields of application, medical imaging is one of those. Today these kind of networks find solutions that are on par or better than many state-of-the-art algorithms [44].

3.4.2. Applications

Deep learning now offers a large set of new tools that are applicable to many problems in the world of medical image processing. Indeed, these tools have already been widely employed. In particular, perceptual tasks are well suited for deep learning.

On the international conference of Medical Image Computing and Computer-Assisted Intervention (MICCAI) in 2018, approximately 70% of all accepted publications were related to the topic of deep learning [45]. Given this fast pace of progress, it is not possible to describe all relevant publications here. Despite this, in the following there is an overview of the most relevant applications like image segmentation, image detection and recognition, image diagnosis, physical simulation and image reconstruction, including one or two significant examples for each category [45, 46].

- **Image segmentation**

Medical image segmentation is a method for dividing an image into multiple regions based on a specific feature. It allows the scientist to focus on an object based on shape, volume, relative position, and abnormality.

Liu et al. [47] built a model that combines a deep CNN (SegNet) with the 3D simplex deformable approach and applied this model of segmentation to images of musculoskeletal tissue in magnetic resonance imaging. Advantages of SegNet are as follows: it is designed for analysis of high-resolution images, which is required in musculoskeletal imaging to show fine details such as thin cartilage. Its scheme provides high memory and efficient computation to reduce output time. It is easy to implement and this enables multiple musculoskeletal applications. The design of 3D simplex deformable modeling preserves information about the shape and surface of musculoskeletal structure. In short, their method produces rapid and accurate results in clinical studies.

- **Image detection and recognition**

Image detection and recognition deals with the problem of identifying medically significant features within an image, for example, tumors, anatomical structures, and cells. In many cases, the images are volumetric. Therefore efficient parsing is a must. A popular strategy to do so is marginal space learning, as it is efficient and allows to detect organs robustly. Its deep learning counter-part is even more efficient, as its probabilistic boosting trees are replaced using a neural network-based boosting cascade. This approach drives efficiency even further by replacing the search process by an artificial agent that follows anatomy to detect anatomical landmarks using deep reinforcement learning. The method is able to detect hundreds of landmarks in a complete CT volume in few seconds.

- **Image diagnosis**

Computer-aided diagnosis is regarded as one of the most challenging problems in the field of medical image processing. The clinicians who interpret medical images can benefit from computer-aided systems that provide more detailed information about certain features that might help to discriminate pathology and aid in treatment planning. There are several applications of medical image diagnosis via DCNN.

The diagnosis of the thyroid nodule is currently based on the analysis of images acquired by ultrasound, a real-time and noninvasive technology, to determine whether the nodule is malignant, indeterminate, or suspicious. Thyroid nodules are heterogeneous in appearance and have many internal components and vague boundaries that make it difficult to differentiate between benign and malignant. To cite a representative example, Ma et al. [48] hybridize two CNNs in order to eliminate operational error and improve the accuracy of the result. The two networks were trained separately and then fused together to diagnose the thyroid nodule based on a softmax classifier. The proposed model results in an accuracy of approximately 83.02%.

- **Physical simulation**

A new field of deep learning is the support of physical modelling. So far this has been exploited in the gaming industry to compute realistically appearing physics engines, or for smoke simulation in real-time.

Unberath et al. [49] propose DeepDRR, a framework for fast and realistic simulation of fluoroscopy and digital radiography from CT scans, tightly

integrated with the software platforms native to deep learning. They use machine learning for material decomposition and scatter estimation in 3D and 2D, respectively, combined with analytic forward projection and noise injection to achieve the required performance. On the example of anatomical landmark detection in X-ray images of the pelvis, the authors demonstrate that machine learning models trained on DeepDRRs generalize to unseen clinically acquired data without the need for re-training or domain adaptation. Their results are promising and promote the establishment of machine learning in fluoroscopy-guided procedures.

- **Image reconstruction**

The reconstruction of an image from the acquired data is an inverse problem. Often, it is not possible to exactly solve the inverse problem directly. In this case, a direct algorithm has to approximate the solution, which might cause visible reconstruction artifacts in the image. Iterative algorithms approach the correct solution using multiple iteration steps, which allows to obtain a better reconstruction at the cost of a higher computation time.

Also the field of medical image reconstruction has been affected by deep learning, a recent paper by Zhu et al. [50] proposes to learn the entire reconstruction operation only from raw data and corresponding images. The basic idea is to model an autoencoder-like dimensionality reduction in raw data and reconstruction domain. Then both are linked using a nonlinear correlation model. The entire model can then be converted into a single network and trained in an end-to-end manner. In the paper, they show that this is possible for 2-D MR and PET imaging and largely outperforms traditional approaches.

3.4.3. Focus on ophthalmology

Also ophthalmology greatly benefited from the recent developments in deep learning. New studies, including pre-registered prospective clinical trials, have shown DL systems are accurate and effective in detecting diabetic retinopathy (DR), glaucoma, age-related macular degeneration (AMD), retinopathy of prematurity, refractive error and in identifying cardiovascular risk factors (e.g. age, blood pressure, smoking status and body mass index, figure 33) from digital fundus photographs. There is also increasing attention on the use of AI and DL systems in identifying disease features, progression and treatment response for retinal diseases such as neovascular AMD and diabetic macular edema using optical coherence tomography (OCT). Additionally, the application of ML to visual fields may be useful in detecting glaucoma progression [51].

Ophthalmology is on the cusp of a revolution in the screening, diagnosis, and management of eye disease. This revolution is being led by computer-based deep learning technology that has the potential to change the practice of ophthalmology. The dependence on imaging makes the field of ophthalmology perfectly suited to benefit from DL algorithms. Incorporation of DL algorithms into the practice of ophthalmology has begun and could potentially change the fundamental type of work performed by ophthalmologists.

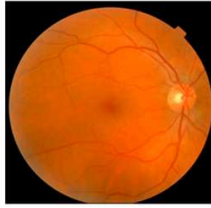
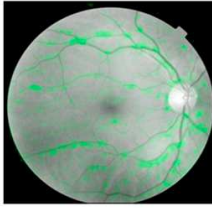


Original	Age	Smoking Status	Systolic BP
			
	Actual: 53.0 years Predicted: 53.8 years	Actual: Nonsmoker Predicted: Nonsmoker	Actual: 128.5 mmHg Predicted: 130.1 mmHg

Figure 33. Attention maps for a single retinal fundus image. The left-most image is a sample retinal image in color. The remaining images show the same retinal image, but in black and white. The soft attention heat map for each prediction is overlaid in green, indicating the areas of the heat map that the neural-network model is using to make that particular prediction for the image.

DL solutions can assist clinicians in identifying abnormalities present via a diagnostic test. In such a scenario, a provisional diagnosis with associated description of key abnormalities found with the test may be provided by computers. Confirmation of the diagnosis as well as counselling and treatment would remain the responsibility of the ophthalmologist. DL diagnostic systems could be integrated into the primary care setting reducing or potentially eliminating unnecessary referrals. Taking such systems one step further, DL tools could enable ophthalmic self-monitoring by patients via smartphone retinal photography, visual acuity and visual field testing. Such technology would empower patients, facilitate early diagnosis, as well as identify treatable eye disease. Consistent interpretation of ocular data by DL might also facilitate high quality ophthalmic research by reducing grading and tester variability. Moreover DL in ocular imaging may be used in conjunction with telemedicine as a possible solution to screen, diagnose and monitor major eye diseases for patients in primary care and community settings especially for those situation where is difficult to have trained ophthalmologist this can really make the difference [52, 53].

Now some significant examples in recent literature to underline how DL could be relevant in this particular field are reported.

Globally, 600 million people will have diabetes by 2040, with a third having DR. Screening for DR, coupled with timely referral and treatment, is a universally accepted strategy for blindness prevention. DR screening can be performed by different healthcare professionals, including ophthalmologists, optometrists, general practitioners, screening technicians and clinical photographers. Nonetheless, DR screening programmes are challenged by issues related to implementation, availability of human assessors and long-term financial sustainability [53]. Over the past few years, DL has revolutionised the diagnostic performance in detecting DR. Using this technique, many groups have shown excellent diagnostic performance. To name one Gulshan et al. [54] in 2 validation sets of 9963 images and 1748 images, at the operating point selected for high specificity, the algorithm had 90.3% and 87.0% sensitivity and 98.1% and 98.5% specificity for detecting referable diabetic retinopathy. These results

suggest that the use of this algorithm could lead to improved care and outcomes compared with current ophthalmologic assessment.

It is projected that 288 million patients may have some forms of age-related macular degeneration (AMD by 2040, with approximately 10% having intermediate AMD or worse [53]. With the ageing population, there is an urgent clinical need to have a robust DL system to screen these patients for further evaluation in tertiary eye care centres. In order to find a solution Lee et al. [55] develop a CNN. Starting from 2.6 million OCT images linked to clinical data points from the electronic medical record (EMR), 52690 normal macular OCT images and 48312 AMD macular OCT images were selected. Then a deep neural network was trained to categorize images as either normal or AMD. At the image level, they achieved an area under the ROC curve of 92.78% with an accuracy of 87.63%. Authors conclude that these findings have important implications in utilizing OCT in automated screening and the development of computer-aided diagnosis tools in the future.

Another challenge in the development of AI models in ophthalmology has been the limited availability of large amounts of data for both the rare diseases and for common diseases which are not imaged routinely in clinical practice. There is a consistent part of literature dedicated to image simulation, a possible answer to this current issue. For example Schiffrers et al. [56] demonstrate that cycleGANs may synthesize virtual angiographic images from their conventional fundus counterparts (figure 34). Fluorescent angiographic methodology augments the capability to image the functional state of retinal circulation of conventional fundus imaging. Despite the diagnostic benefits, physicians are increasingly reluctant to use angiographic imaging technology because of its severe potential side effects. A successful synthesization of an angiographic image could reduce the need for actual angiographic imaging. Moreover it could allow to create large artificial dataset, one of the key point to efficiently train modern algorithms.

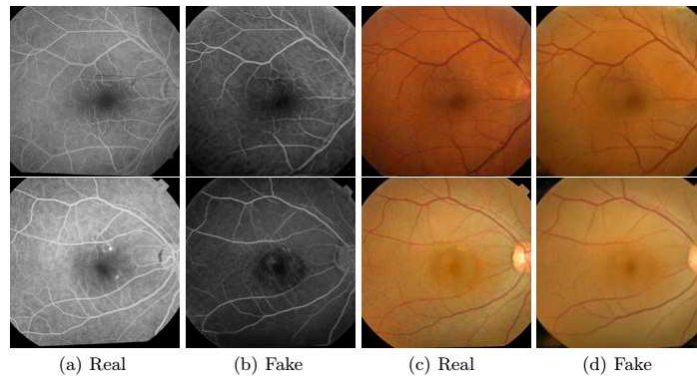


Figure 34. Each row shows from left to right the real and generated angiographic image, the authentic color image and the reconstructed color image to show cycle consistency.

3.5. Related work

Studies on automatic cataract diagnosis have gained wide attention for years, owing to the great harm of cataract. Even if it is not the only kind of image uses for this task, this paragraph mainly focuses on fundus images. Recently many methods have been proposed to solve this issue and they are manly based on either machine learning or deep learning.

Some exceptions exist, in 2008 Abdul-Rahman et al. [57] proposed a technique to detect cataract based on Fourier transform of fundus images. The idea is that the image spectrum falls away to higher frequencies more quickly in cataractous images with respect to physiological ones. This is due to the less defined image contours in pathological fundus images and the authors found a parameter that describes this behaviour and useful for cataractous image classification. Some years later Zheng et al. [58] used a similar approach exploiting Fourier transform. Once calculated power spectrum, linear discriminant analysis (LDA) performs cataract grading into two and four level according to the disease severity.

Following researches tend to classify fundus images not only in healthy and pathological, they propose a more specific classification including various stages of the disease. Usually authors classify these images as non-cataractous, mild, moderate and severe cataract.

In [59], Guo et al. examined wavelet transform as well as sketch-based methods to extract features from fundus images. After the integration of these features they applied multi-Fisher classification algorithm for cataract detection and grading. To improve the accuracy of diagnosis of cataract, Yang et al. [60] proposed an ensemble learning based approach. In this approach, three independent feature sets: wavelet-, sketch-, and texture-based features, which were extracted from fundus images, were served to build SVM models and Back Propagation Neural Network models; after that, majority voting and stacking were utilized to integrate the multiple base learning models for final fundus image classification. Another difference between the two approaches is the use of pre-processing. Guo et al. chose raw images while Yang et al. transformed fundus images before using them as input. In particular authors converted original images from RGB color space to the green channel and after that they applied histogram equalization to further increase the global contrast.

These kinds of pre-processing are widely used in cataract detection algorithm because it has been seen that enhance the contrast between the background and blood vessels improves classification performances. In addition to histogram equalization Kolhe et al. [61] performed skeletonization to reduce foreground regions in a binary image and highlighted vessel pathways. After that authors designed two different algorithms: a binary SVM to classify fundus images and a multi-class Fisher discriminant analysis algorithm (MDA) to grade cataract images into mild, moderate and severe.

SVM approach is very popular in literature as machine learning tool and it is used to solve a wide range of tasks. Also Harini et al. [62] based their algorithm on an SVM approach. A significative difference from the previous one is that in [62] authors wanted to improve the quality of the images. A high-quality image dataset can bring significant improvement in final results. In order to remove gaussian noise a mean filter was implemented. Authors said

that other filters involved in salt and pepper noise reduction are not necessary since fundus images were captured by high quality fundus cameras. It is worth mentioning Qiao et al. [63] and their SVM approach, in this research authors chose to extract features not from the whole image but from sub-images. The algorithm first divides the whole image into 16 small-block images evenly. Since the optic disc is an important basis for detecting the cataract, the sub-image containing the optic disc is taken out separately when the image is divided into blocks. After that, features (such as color feature, wavelet feature and texture feature) are extracted from each sub-image and finally support vector machine is used to train and classify fundus images. This approach reveals one of the best performances compared to the others machine learning approaches previously analyzed. Indeed it has an accuracy equal to 95.33% when classify images as cataractous or not and if it grades fundus images into four level it reaches more than 87%.

A quite different approach was proposed by Song et al. [64]. The authors utilized semi-supervised learning for automatic cataract classification and grading. They decided to use a large amount of unlabeled images together with a small part of labeled ones. In the stage of building a classifier, two base algorithms are used which are Bayesian network and decision tree, respectively. Then they used tri-training which generates three classifiers from the original labeled examples and finally unlabeled examples refine initial classifier in an iterative method.

So far, only machine learning based examples have been discussed, however a consistent part of literature regards deep learning approach seems to be very promising. In the field of automatic cataract detection one of the first deep learning-based algorithm dates back to 2013. Yang et al. [65] built a three-layer back propagation network with 40 neurons within the input layer. Each of these neurons represents one of the 40 features extracted from luminance features, gray co-occurrence matrix and gray-gradient co-occurrence matrix.

The idea to use features instead of images as input it is not a common choice for this kind of classification task, in fact more recent researches focus on convolutional neural network, obtaining, in general, better results. The deep convolutional neural network of Zhang et al. [66] accepts as input pre-processed fundus images where only green channel is displayed. Authors trained this network with a dataset containing more than 5600 images. A clear difference between articles about machine learning and the ones about deep learning is the amount of samples within the respective datasets. Talking about hundreds of images for machine learning and thousands for deep learning, in general, it is an order of magnitude the difference between them. It is well known that to efficiently train a deep neural network big datasets are necessary. Moreover, since tests are performed in a larger amount of data, this characteristic inevitably improves classification reliability.

Dong et al. [67] started their study from a CNN utilized only for feature extraction. The final classification step of classification is reserved to either an SVM algorithm or a Softmax approach. They compared these two techniques in order to see which one has the best results. In their case Softmax turns out to have better accuracies both for two- and four-level grading.

In [68], Li et al. proposed a ResNet of 18-layer that inputs retinal fundus images once again in green channel. The interesting point of this article is that the algorithm produces as output also heatmaps. Heatmaps highlight

the regions in an image that the CNN focuses on while trying to make a prediction. These results confirm that optic disc and blood vessels are important features in cataract diagnosis. Zhang et al. [69] wanted to classify fundus images into six different level of cataract severity, including non-cataractous, slightly mild, mild, medium, slightly severe and severe cataract. This ambitious target finds his basements on a complex classification system. The algorithm extracts high-level features from residual network and texture features from gray level co-occurrence matrix. Then to automatically grade cataract two support vector machine (SVM) classifiers are used as base-learners to obtain the probability outputs of each fundus image. Last step: fully connected neural network (FCNN) is used as meta-learner to output the final classification result, which consists of two fully-connected layers. The results are interesting, the whole process has an average accuracy of 92.66% in six-level grading.

One of the most recent research conducted in this field was published in 2019 by Xu et al. [70]. Similar to [63] authors divided input images into eight squared sub-images that cover the whole retina. After that they built a CNN that separately analyze each sub-image. Finally a majority voting approach is performed in order to classify original fundus images.

It is worth mentioning one last thing. Each dataset of all these studies were built starting from high-quality fundus images captured either in hospitals or specialized clinics. Physicians and eye specialist ensure the quality and the labels reliability of the images and this is a fundamental prerogative to start any automatic diagnose research.

4. Method

This chapter presents the main part of the thesis project. The aim is to develop an algorithm able to automatically classify a D-EYE fundus image in according to the presence of cataract or not. D-EYE, of which can be found a more detailed description in section 2.4, is a smartphone-based digital direct ophthalmoscope that produces retinal videos and photos.

This device is particularly suitable for screening operation. Some algorithms have already been designed for D-EYE outputs. Currently it can extrapolate from retinal fundus videos the "good" and the "bad" frames. Good frames are the ones where the retina is clearly visible while bad frames are all the others, where retina is unfocused or not present. Indeed, lot of frames result useless for eye examination, such as the ones show exam room details, patient faces or eyelids when subjects blink. If necessary, frame selection can be more specific by picking only images where optic disc is present and focused. From these frames, the algorithm is then able to perform optic disc and cup segmentation. This procedure could be very useful for physicians, indeed the cup-to-disc ratio (often notated CDR) is a key parameter that ophthalmologists use for glaucoma detection and for monitoring how the disease advances.

Now the company aims to broaden the range of possible applications of his device. D-EYE wants to investigate if automatic algorithms can provide a reliable support to the diagnoses of various eye disease including cataract. Automation is very important for this kind of medical device whose target is screening. Thanks to this improvement also non-expert ophthalmologists will be able to use D-EYE, make diagnoses and share hypothesis with other physicians to confirm results. It is worth pointing out that the algorithm can not replace years of study and ophthalmologists professionalism. Rather than an alternative it has to be seen as a supporting tool.

The next paragraphs will be organized as follow. First it describes the two image datasets used to build the CNN and the different image pre-processing utilized to improve classification accuracy. Finally the last three paragraphs present an overview of all phases necessary to implement the convolutional neural network. The first one outlines the internal structure of the network while the following two summarize training and testing processes, respectively.

4.1. Dataset

Building an effective dataset is the first step towards creating an accurate convolutional neural network. As said in 3.3, the training phase of a CNN is dataset-dependent, in particular the amount and the quality of the images greatly affect final results.

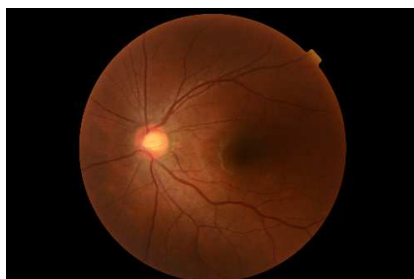
4.1.1. Dataset 1 - ODIR5k and Retina dataset

Initially, in order to create a satisfactory starting point, two different public datasets of fundus oculi images are merged. A description of both sources will be now presented.

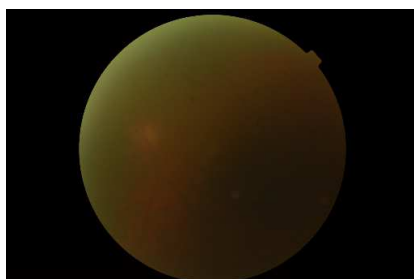
Retina dataset (https://github.com/yiweichen04/retina_dataset)

It contains 601 fundus oculi images in total. Each image is labelled in accordance with one of the following categories:

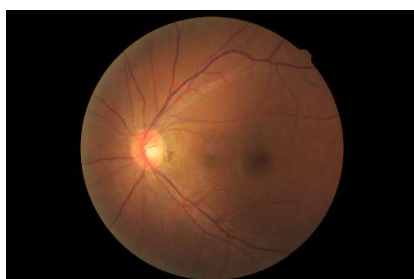
- *Normal*, without any disease. 300 images. Example of file name categorized as Normal: “NL_001”.



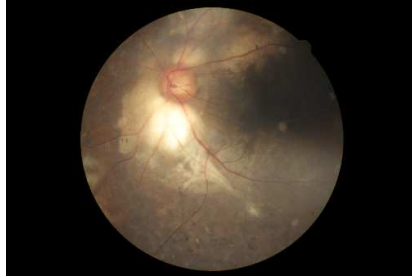
- *Cataract*, affected by cataract. 100 images. Example of file name categorized as Cataract: “Cataract_001”.



- *Glaucoma*, affected by glaucoma. 101 images. Example of file name categorized as Glaucoma: “Glaucoma_001”.



- *Retina*, affected by other retina disease different from either cataract or glaucoma. 100 images. Example of file name categorized as Retina: “Retina_001”.



Regarding image resolution, inside this dataset there are three different pixel sizes:

- 1848x1224: 40 images (Normal 22, Cataract 5, Glaucoma 5, Retina 8)
- 2464x1632: 158 images (Normal 81, Cataract 19, Glaucoma 22, Retina 36)
- 2592x1728: 403 images (Normal 197, Cataract 76, Glaucoma 74, Retina 56)

No information about how these images were collected is available. Since the target of this work is to diagnose specifically cataract and not other eye disease to build the final dataset only *Normal* and *Cataract* images were considered and taken into account.

Ocular Disease Recognition – ODIR5K (<https://odir2019.grand-challenge.org/>)

This is a wider and more complicated dataset than the previous one. It is a structured ophthalmic database of 5000 patients with age, colour fundus photographs from left and right eyes and diagnostic keywords from doctors. Since this dataset is part of an International Competition on Ocular Disease Intelligent Recognition (ODIR), hosted by Peking University (PKU) and organized by National Institute of Health Data Science at Peking University (NIHDS-PKU) and Institute of Artificial Intelligence at Peking University (IAI-PKU), the dataset is divided into training set (7000 images), with annotation, and testing set (1000 images), without annotation. From now on only images belong to training set will be considered.

For each patient annotations are labeled by trained human readers with quality control management. They classify subject into eight labels including normal (N), diabetes (D), glaucoma (G), cataract (C), age-related macular degeneration (A), hypertension (H), myopia (M) and other diseases/abnormalities (O) based on both eye images and additionally patient age. Moreover, for each fundus image there is a diagnostic keyword which described the clinical situation of that particular eye. The following list summarize all the different diagnostic keywords used in this dataset:

- Anterior segment image
- asteroid hyalosis
- atrophy
- branch retinal artery occlusion
- branch retinal vein occlusion
- cataract
- central retinal vein occlusion
- chorioretinal atrophy
- depigmentation of the retinal pigment epithelium

- diabetic retinopathy
- drusen
- dry age-related macular degeneration
- epiretinal membrane
- fundus laser photocoagulation spots
- glaucoma
- hypertensive retinopathy
- idiopathic choroidal neovascularization
- image offset
- laser spot
- lens dust
- low image quality
- macular coloboma
- macular epiretinal membrane
- macular hole
- maculopathy
- mild non proliferative retinopathy
- moderate non proliferative retinopathy
- myelinated nerve fibers
- myopic maculopathy
- normal fundus
- old branch retinal vein occlusion
- old central retinal vein occlusion
- old chorioretinopathy
- old choroiditis
- optic disc edema
- optic disk epiretinal membrane
- optic nerve atrophy
- pathological myopia
- peripapillary atrophy
- pigment epithelium proliferation
- post laser photocoagulation
- post retinal laser surgery
- proliferative diabetic retinopathy
- punctate inner choroidopathy
- refractive media opacity
- retinal pigmentation
- retinochoroidal coloboma
- rhegmatogenous retinal detachment
- severe non proliferative retinopathy
- severe proliferative diabetic retinopathy
- silicone oil eye
- spotted membranous change
- suspected glaucoma
- suspected moderate non proliferative retinopathy
- suspected retinal vascular sheathing
- tessellated fundus
- vitreous degeneration
- wedge white line change
- wet age-related macular degeneration
- white vessel

In the same annotation image it is possible to find one or more diagnostic keywords. Figure 35 presents a complete example of a patient description.

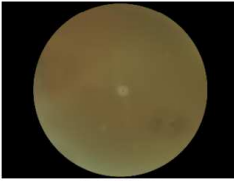
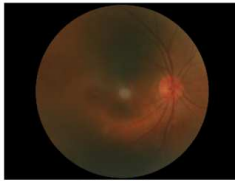
Basic Info.	<i>Patient Sex</i>	Female			<i>Patient Age</i>	69		
Fundus Images								
	0_left.jpg				0_right.jpg			
Laterality	Left				Right			
Disease Labels	<i>N</i>	<i>D</i>	<i>G</i>	<i>C</i>	<i>A</i>	<i>H</i>	<i>M</i>	<i>O</i>
	0	0	0	1	0	0	0	0
Diagnostic Keywords	Cataract				Normal fundus			

Figure 35. Example of structured ophthalmic record in ODIR-5K dataset.

As said before, the focus is mainly on cataract images, for this reason ODIR-5K can be simplified as follow:

- *Not cataract*, 6707 images (normal fundus 2870, other diagnosis different from cataract 3837)
- *Cataract*, 293 images (only cataract 249, cataract and other diagnostic keywords 44)

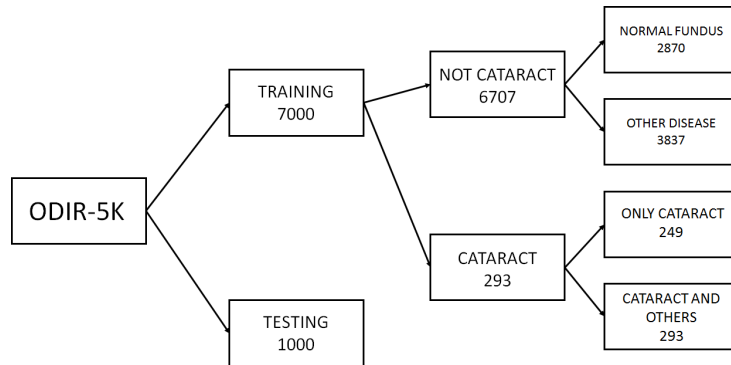


Figure 36. Simplified structure of ODIR-5K.

Regarding image resolution, inside this dataset there are several different pixel sizes, it is possible to divide them into three different intervals:

- 250x188 - 972x785: 41 images
- 1280x960 - 1974x1483: 1547 images
- 2048x1536 - 5184x3456: 5412 images

This dataset is “real-life” set of patient information collected by Shanggong Medical Technology Co., Ltd. from different hospitals/medical centers in China. In these institutions, fundus images are captured by various cameras in the market, such as Canon, Zeiss and Kowa, resulting into varied image resolutions. Patient identifying information are all removed.

Both sources contain several fundus images acquired by different devices. Each fundus camera has his own properties and produces slightly different images. Moreover figure 37 compares two fundus images, one captured by a fundus camera (figure 37a) and the other captured by D-EYE (figure 37b). As it is possible to notice several differences characterize each image. On the left it is possible to observe an image that has a larger field of view (FoV) and contains almost the entire fundus oculi while one the right D-EYE produces an image, whose FoV is less wide, focuses on the optic disk. In general D-EYE can record images of all parts of the fundus but the most interesting frames regard optic disk and the areas nearby. Another peculiarity belonging only to D-EYE images is the presence of the iris while in fundus camera images usually there is nothing but the fundus oculi.

From now on, for simplicity, the dataset presented in this paragraph will be considered as "dataset 1" while the dataset described in the next paragraph as "dataset 2".

In practice dataset 1 contains several fundus images but these are not comparable to the ones related to the classification target. Even if this dataset remains essential for an initial tuning of the hyperparameters a dataset with D-EYE images is necessary. D-EYE is a relatively young device and it has not

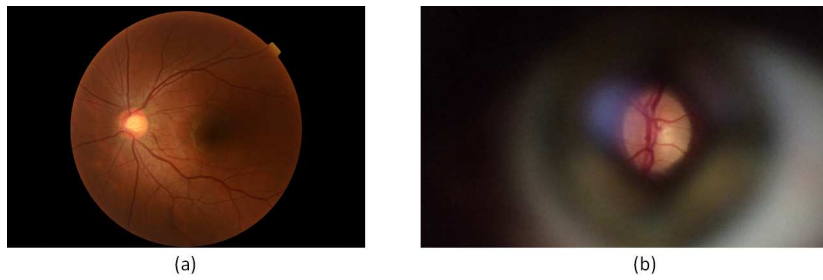


Figure 37. Comparison between fundus images captured by a fundus camera (a) and captured by D-EYE (b).

the possibility to perform a sufficient amount of examination in patient with cataract in order to build a dataset able to train a CNN. This is the reason why the company develop an algorithm that can produce images similar to the ones acquired by D-EYE. Dataset 2 is built thanks to this algorithm.

4.1.2. Dataset 2 - DEYE-like Tool

DEYE-like Tool is the name of the algorithm that modifies fundus camera images to make them similar to those acquired with the D-EYE device. Figure 38 summarizes the main steps to produce DEYE-like images. Since the starting point of this process is a fundus image, dataset 1 is used to extrapolate "raw material". For each image, the software identifies the optic disc and performs a resizing based on the selected area. After that, the code modifies image properties (blur and focus) in order to be consistent with D-EYE images. A set of cropped images is sampled, with each cropped image corresponding to a different location. Last two steps consist in simulating a proper field of view and adding a synthetic iris.

As it is shown in figure 38 (top-right) the results of this process are DEYE-like images. Similar to those that can be extracted from a D-EYE video.

It is worth mentioning that this process has two variants based on the purpose of the generated images. Specifically, if images are designed to be part of the training set then the areas where the algorithm can sample from are uniformly distributed within the ideal square inscribed the retina (figure 39a). Talking about testing images, the software selects a more restricted zone, considering only a circumference that includes the OD (figure 39b).

Thanks to this code it is possible to generate an infinite number of samples to train and test networks. For this project three different datasets are created, one for training and validation and the other two only for testing.

- Dataset 2.1: 6778 images (Non cataract 5544, Cataract 1234) utilized for training and validation.
- Dataset 2.2: 3506 images (Non cataract 2367, Cataract 1139) utilized for testing.
- Dataset 2.3: 1551 images (Non cataract 189, Cataract 1362) utilized for testing.

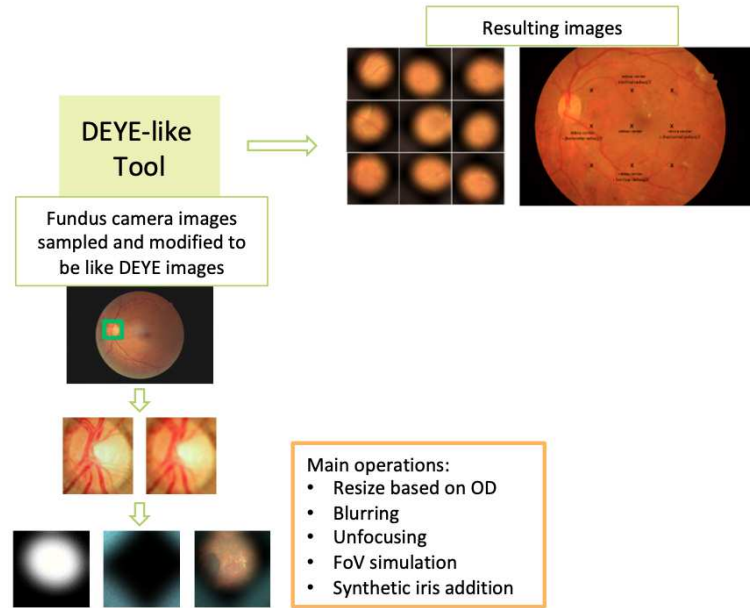


Figure 38. DEYE-like Tool principle of working.

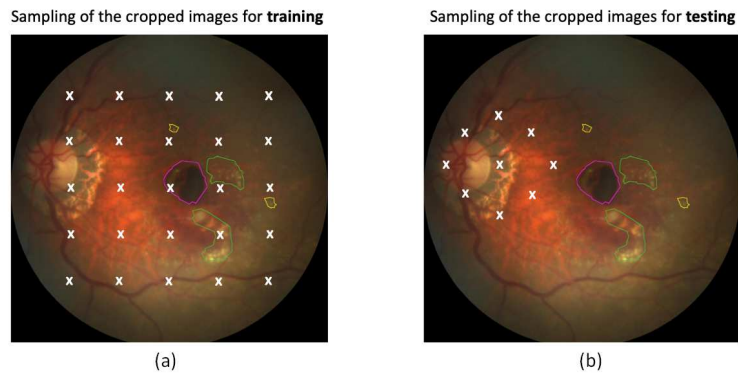


Figure 39. Comparison between the processes of sampling in images related to training and testing.

4.2. Pre-processing

In general, CNNs are designed to extrapolate important features directly from input images. For this reason, sophisticated pre-processing is usually not necessary. However, for this project, better performances are reached using modified images instead of raw ones as input because the literature review shows that some specific elaboration could improve both training phase and classification performances. First each image has been resized to a smaller resolution in accordance to input size of the different CNNs implemented. This size reduction is necessary to allow the net to process images but also it standardizes all inputs

and significantly reduces computational time.

Taking into account a smaller amount of pixels could be risky for the loss of information. Therefore, image dimensions have been chosen to have a good compromise between preserving all important features (such as optic disc, vessels and macula) and having an acceptable computational time.

Several pre-processing approaches for fundus images can be found in literature. Each of them is thought to enhance the visualization of a certain feature. For example, considering an RGB image, in order to obtain a better contrast between the background and blood vessels many research suggest looking at green channel. Instead, if the target is optic disc detection sometimes red channel appears to be a better choice [71]. The following paragraphs will present the different pre-processing procedures tried in order to improve classification accuracy.

4.2.1. Fourier Transform

The Fourier transform is an operation that transforms data from the time (or spatial) domain into the frequency domain. Beside several application in signal processing field, Fourier Transform is an important image processing tool which is used to decompose an image into its sine and cosine components. The output of the transformation represents the image in the Fourier or frequency domain, while the input image is the spatial domain equivalent. In the Fourier domain image, each point represents a particular frequency contained in the spatial domain image.

Since this work concerns only with digital images, the following description will be restricted only to Discrete Fourier Transform (DFT). The DFT is the sampled Fourier Transform and therefore does not contain all frequencies forming an image, but only a set of samples which is large enough to fully describe the spatial domain image. The number of frequencies corresponds to the number of pixels in the spatial domain image, i.e. the image in the spatial and Fourier domain are of the same size.

$$F(k, l) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i, j) e^{-i2\pi(\frac{ki}{N} + \frac{lj}{N})}$$

where $f(i, j)$ is the image in the spatial domain and the exponential term is the basis function corresponding to each point $F(k, l)$ in the Fourier space. The basis functions are sine and cosine waves with increasing frequencies, $F(0,0)$ represents the DC-component of the image which corresponds to the average brightness and $F(N-1, N-1)$ represents the highest frequency.

In most implementations, and also in this work, the Fourier image is shifted in such a way that the DC-value, $F(0,0)$, is displayed in the center of the image. The further away from the center an image point is, the higher is its corresponding frequency. Moreover, since the dynamic range of the Fourier coefficients is too large to be displayed on a image, it is necessary to apply a logarithmic transformation in order to build a figure that allows to distinguish the components of all frequencies.

The use of Fourier transforms to analyse the optical property of images is well known. It has been used to examine ocular optical quality, analysis of ocular dynamic wavefront aberrations, early diagnosis of glaucoma, automated localization of anatomic features on the retina, modelling polar variations in videokeratographic power values and assessment of corneal endothelial cell structure. As introduced in section 4.1 both in [57] and [59] authors used discrete Fourier transform to pre-process images. This reveals that also in cataract detection is a well-established technique.

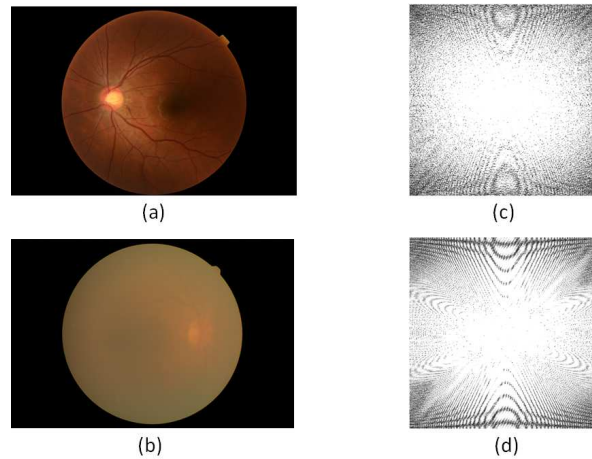


Figure 40. Example of image pre-processing with Fourier transform. Figure (a), (b): original fundus images cataractous and not cataractous, respectively. Figure (c), (d): images (a) and (b) after Fourier transform, respectively.

4.2.2. Green channel

An RGB image can be considered as a 3D array with dimensions m -by- n -by-3. The first two elements, m and n , indicate width and height while the third element specifies color channel: red, green or blue. Therefore every pixel expresses, for each color, a level of brightness, usually within the interval 0-255. Taking into account only one channel means creating a new image using only the matrix of pixel related to a color, in this case green.

This kind of image pre-processing is very popular when retinal images are involved. Several researches [58][60][64][65][66][68][70] exploit green channel in order to enhance contrast between the background and blood vessels and to reduce artifacts due to uneven illumination. Another interesting aspect of using only one color channel instead of three is that this reduces the amount of data by $2/3$ archiving an effective data compression and greatly reducing the computational time.

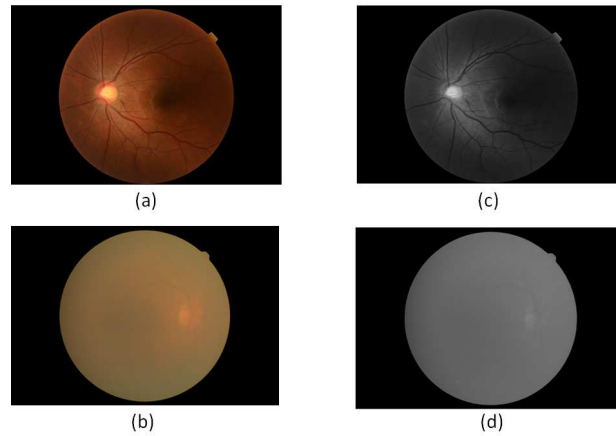


Figure 41. Example of image pre-processing with green component. Figure (a), (b): original fundus images cataractous and not cataractous, respectively. Figure (c), (d): images (a) and (b) after extrapolating only green channel, respectively.

4.2.3. Histogram equalization

Histogram equalization is a more sophisticated technique, it modifies the dynamic range of an image by altering the pixel intensity values guided by the histogram of that image. Recalling that the intensity histogram of an image is a table of counts, each bin represents a range of intensity values. The counts record the number of times each intensity value range occurs in the image. For an RGB image, there is a separate table entry for each of the R, G, and B components. Histogram equalization creates a non-linear mapping, which reassigns the intensity values in the input image such that the resultant images contain a uniform distribution of intensities, resulting in a flat (or nearly flat) histogram. The resulting image typically brings more image details to light, since it makes better use of the available dynamic range [72].

It is not difficult to find image analysis studies where this technique is used to enhance general contrast. Regarding fundus images, histogram equalization is usually applied [60][64][65] together with the isolation of the green component, while in [61] it is used to uniformly adjust the contrast of the image before applying skeletonization.

In order to perform histogram equalization in fundus images it is possible to exploit two different MATLAB functions.

- $J = \text{histeq}(I, hgram)$ transforms the RGB image I so that the histogram of the output RGB image J with length bins approximately matches the target histogram $hgram$.
- $J = \text{adapthisteq}(I)$ enhances the contrast of the grayscale image I by transforming the values using contrast-limited adaptive histogram equalization (CLAHE)

CLAHE is an adaptive contrast enhancement method. It is based on adaptive histogram equalization (AHE), where the histogram is calculated for the contextual region of a pixel. The pixel's intensity is thus transformed to a value

within the display range proportional to the pixel intensity's rank in the local histogram. CLAHE is a refinement of AHE where the enhancement calculation is modified by imposing a user-specified maximum, ie, clip level, to the height of the local histogram, and thus on the maximum contrast enhancement factor. The enhancement is thereby reduced in very uniform areas of the image, which prevents overenhancement of noise and reduces the edge-shadowing effect of unlimited AHE. [73]

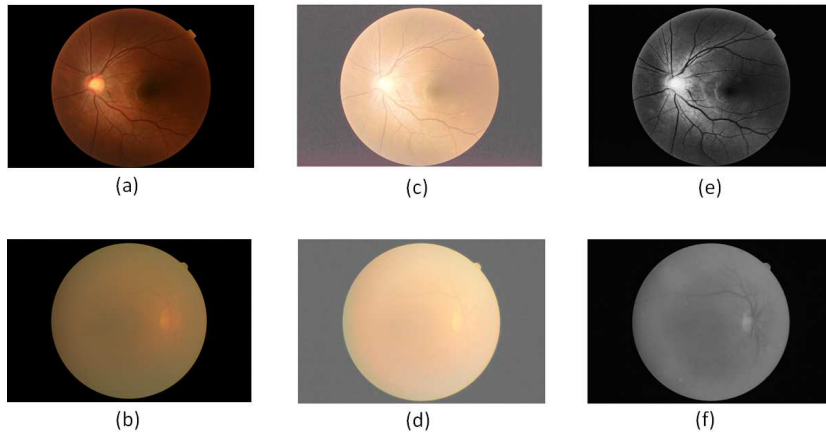


Figure 42. Example of image pre-processing with histogram equalization. Figure (a), (b): original fundus images cataractous and not cataractous, respectively. Figure (c), (d): images (a) and (b) after using *histeq*, respectively. Figure (e), (f): images (a) and (b) after using *adaphisteq*, respectively.

4.3. CNN architecture

The core of the algorithm is the network architecture, i.e. how each layer is implemented and how all layers are chained together. A first approach to achieve the task of this work is using transfer learning. Paragraph 3.3.4 offers a brief introduction to this technique and describes his major advantages.

MATLAB offers an easy way to implement transfer learning and it provides ready-to-load the most common CNNs such as AlexNet, GoogLeNet, SqueezeNet and others. Listing 1 shows an example of the code utilized to create a network starting from AlexNet.

AlexNet is a convolutional neural network that has 25 layers. The network it is possible to load from MATLAB is a pretrained version of the network trained on more than a million images from ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images.

Listing 1. Example of transfer learning code

```
1 net = alexnet;
```

```

2 layersTransfer = net.Layers(1:end-3);
3
4 samples = imdsTrain.countEachLabel;
5 classWeights = 1./samples.Count;
6 classWeights = classWeights'/mean(classWeights);
7
8 layers = [
9     layersTransfer
10
11     fullyConnectedLayer(2, 'WeightLearnRateFactor',20,
12     'BiasLearnRateFactor',20)
13     softmaxLayer
14     weightedClassificationLayer(classWeights)];

```

All the layers but the last three are transferred to the new network. The replacement of the last layers is necessary to let the neural network be able to classify retinal images as request by the task. As it is possible to notice last layers are strictly related to the final classification. Paragraphs 3.2.4 and 3.2.5 describe *fully connected layer* and *softmax layer*, respectively. Here it is worth presenting some more details about input options in the fully connected layer and what is a *weighted classification layer*.

Looking at the code, fully connected layer is called in line 11. Value 2 indicates the number of classes while *WeightLearnRateFactor* and *BiasLearnRateFactor* are nonnegative scalar numbers by which the software multiplies the global learning rate to determine the learning rate for the weights in this layer.

Weighted classification layer is a particular type of classification layer useful in cases, such as this task, where there the dataset is unbalanced. In general classification layer computes the cross entropy loss for multi-class classification problems with mutually exclusive classes. The peculiarity of this layer, as suggests the name, is that the loss is calculated setting the weights in according to the number of samples in each class. This improves the network ability to correctly classify images belonging to the class with less samples, the pathological one. This is the most critical classification since it is fundamental to lower cataract false negative and weighted classification layer is a good strategy to reduce this problem.

Starting with a pre-trained network is ideal to test the feasibility of the project. It is a relatively quick way to see if the accuracy reached is sufficient to encourage further studies with deep learning. On the other hand, once performed this first step, a more specific network is necessary. AlexNet is specialized in classify common objects like pencil, keyboards and several type of animals. For this reason it is important to build a network ex novo, specific for retinal images. The proposed structure is the following one (Listing 2).

Listing 2. CNN architecture implemented

```

1 inputSize = [320 320 3];
2
3 samples = imdsTrain.countEachLabel;
4 classWeights = 1./samples.Count;
5 classWeights = classWeights'/mean(classWeights);

```



```

6
7 cnn = [
8     imageInputLayer(inputSize)
9
10    convolution2dLayer(3,8,'Padding','same')
11    batchNormalizationLayer
12    reluLayer
13
14    maxPooling2dLayer(2,'Stride',2)
15
16    convolution2dLayer(3,16,'Padding','same')
17    batchNormalizationLayer
18    reluLayer
19
20    maxPooling2dLayer(2,'Stride',2)
21
22    convolution2dLayer(3,32,'Padding','same')
23    batchNormalizationLayer
24    reluLayer
25
26    maxPooling2dLayer(2,'Stride',2)
27
28    convolution2dLayer(3,32,'Padding','same')
29    batchNormalizationLayer
30    reluLayer
31
32    fullyConnectedLayer(10)
33    fullyConnectedLayer(2)
34    softmaxLayer
35    weightedClassificationLayer(classWeights)];

```

Image input size (320x320) is chosen to be a submultiple of usual dimensions of images acquired by D-EYE (640x640). From line 7 to line 35 it is possible to observe the whole CNN structure. This network consists in 20 layers in total, the proposed architecture consists of four convolutional units interspersed with max pooling layers. A convolutional unit includes a convolutional layer, a batch normalization layer and a ReLU layer.

In each convolutional layer the parameter *FilterSize* is 3, this means that the size of the local regions to which the neurons connect in the input image is 3x3. Regarding *NumFilters*, they are set to progressively larger numbers (8-16-32), the deeper the algorithm goes into the network the higher is the number of neurons in the convolutional layer that connect to the same region in the input. The *Stride* remains 1 by default and *Padding, same* indicates that the software calculates the size of the padding at training time so that the output has the same size as the input. Figure 43 shows the outputs after the input image passes through the first convolutional layer. Once again optic disc and blood vessels highlight the difference between the non cataractous image (figure 43a) and the cataractous one (figure 43b) and this strengthen the importance of these features.

The second layer of these convolutional units is a batch normalization layer.

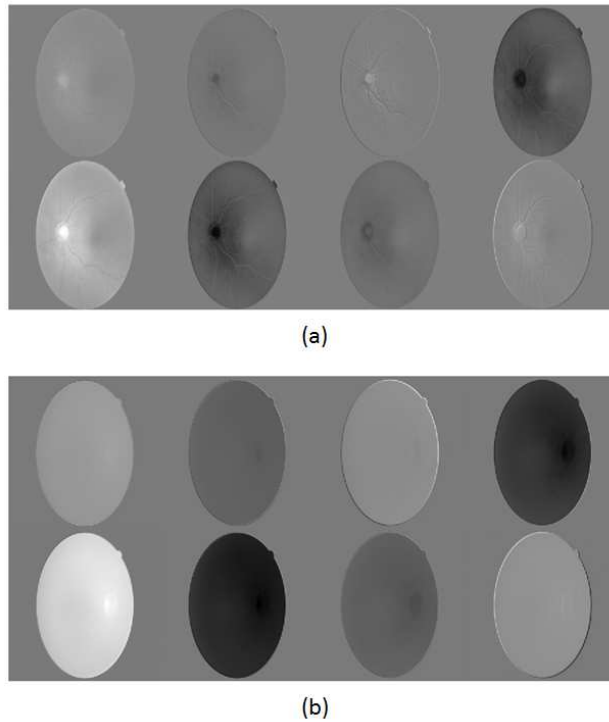


Figure 43. Output images after the first convolutional layer. Figure (a) shows the output when the input is a non cataractous image. Figure (b) shows the output when the input is a cataractous image.

Batch normalization layers normalize the activations and gradients propagating through a neural network, making network training an easier optimization problem. To speed up training of convolutional neural networks and reduce the sensitivity to network initialization, batch normalization layers are usually put between convolutional layers and nonlinearities, such as ReLU layers. In practice this kind of layer normalizes its inputs x_i by first calculating the mean μ_B and variance σ_B^2 over a mini-batch and over each input channel. Then, it calculates the normalized activations as

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

Here, ϵ improves numerical stability when the mini-batch variance is very small. To allow for the possibility that inputs with zero mean and unit variance are not optimal for the layer that follows the batch normalization layer, the batch normalization layer further shifts and scales the activations as

$$y_i = \gamma \hat{x}_i + \beta$$

Where the offset β and scale factor γ are learnable parameters that are updated during network training.

The last layers involved into the final classification process are the same as in Listing 1. For a more detailed description of CNN layers refer to section 3.2.

4.4. CNN training

Once tested the workability of the task through various pre-trained networks and defined the definitive CNN architecture, the natural following step is to train the network. This is one of the most critical part and from the correct setting of the hyperparameters greatly depends final results.

The following figure (figure 44) summarizes all the options utilized to train the proposed network and the next part will briefly describe the most important training properties.

```

TrainingOptionsSGDM with properties:

    Momentum: 0.9000
    InitialLearnRate: 1.0000e-04
    LearnRateScheduleSettings: [1x1 struct]
    L2Regularization: 1.0000e-04
    GradientThresholdMethod: 'l2norm'
    GradientThreshold: Inf
    MaxEpochs: 15
    MiniBatchSize: 10
    Verbose: 0
    VerboseFrequency: 50
    ValidationData: [1x1 matlab.io.datastore.ImageDatastore]
    ValidationFrequency: 3
    ValidationPatience: Inf
    Shuffle: 'every-epoch'
    CheckpointPath: ''
    ExecutionEnvironment: 'auto'
    WorkerLoad: []
    OutputFcn: []
    Plots: 'training-progress'
    SequenceLength: 'longest'
    SequencePaddingValue: 0
    DispatchInBackground: 0

```

Figure 44. CNN training options.

The top left caption says: "TrainingOptionsSGDM". SGMD (*Stochastic Gradient Descent with Momentum*) The word "stochastic" means a system or a process that is linked with a random probability. Hence, in Stochastic Gradient Descent, a few samples are selected randomly instead of the whole dataset for each iteration. In Gradient Descent there is a term called "batch" which denotes the total number of samples from a dataset that is used for calculating the gradient for each iteration. In typical Gradient Descent optimization the batch is taken to be the whole dataset. Although using the whole dataset is useful for getting to the minima in a less noisy and less random manner, the problem arises when datasets gets big, such as in this project.

This problem is solved by Stochastic Gradient Descent. In SGD, the algorithm uses only same samples (mini-batch) to perform each iteration. Samples are randomly shuffled and selected for performing each iteration. Since the images from the dataset are chosen at random, the path taken by the algorithm to reach the minima is usually noisier than typical Gradient Descent algorithm.

Even though this process requires a higher number of iterations to reach the minima than typical Gradient Descent, it is still computationally much less expensive than classic Gradient Descent techniques. Hence, in most scenarios, SGD is preferred for optimizing a learning algorithm.

The standard gradient descent algorithm updates the network parameters (weights and biases) to minimize the loss function by taking small steps at each iteration in the direction of the negative gradient of the loss,

$$\theta_{l+1} = \theta_l - \alpha \nabla E(\theta_l)$$

where l is the iteration number, $\alpha > 0$ is the learning rate, θ is the parameter vector, and $E(\theta)$ is the loss function. The stochastic gradient descent algorithm can oscillate along the path of steepest descent towards the optimum. The "M" of SGMD indicates the *momentum*, as shown in figure this parameter is set to 0.9. Adding this term to the parameter update is one way to reduce this oscillation. The stochastic gradient descent with momentum update is

$$\theta_{l+1} = \theta_l - \alpha \nabla E(\theta_l) + \gamma(\theta_l - \theta_{l-1})$$

where γ determines the contribution of the previous gradient step to the current iteration.

InitialLearnRate equals to 0.0001 corresponds to α in the previous formula at the beginning of the process. With the parameter *LearnRateScheduleSettings* it is possible to change, usually decrease, α during the training. In this case the choice is to keep this value constant. Inside his documentation, MATLAB remembers that if the learning rate is too low, then training takes a long time. If the learning rate is too high, then training can reach a suboptimal result. The setting of this parameter aims to have a good compromise between time consuming and optimal results.

Adding a regularization term for the weights into the loss function $E(\theta)$ is one way to reduce overfitting. The regularization term is also called weight decay and in this project is *L2Regularization*. In this way the loss function with the regularization term takes the form

$$E_R(\theta) = E(\theta) + \lambda \Omega(w)$$

where w is the weight vector, λ is the regularization factor (coefficient), and the regularization function $\Omega(w)$ is

$$\Omega(w) = \frac{1}{2} w^T w.$$

GradientThresholdMethod and *GradientThreshold* are parameters related to gradient clipping. If the gradient increases in magnitude exponentially, then the training is unstable and can diverge within a few iterations. This "gradient explosion" is indicated by a training loss that goes to NaN or Inf. Gradient clipping helps prevent gradient explosion by stabilizing the training at higher learning rates and in the presence of outliers.

Epochs and mini-batch are described in section 3.3. After several trials 15 epochs and 10 as mini-batch size reveal good performances together with an acceptable computational time.

VerboseFrequency indicates the number of iterations between of printing onto the command window training progress. This property only has an effect when *Verbose* value equals true therefore in this case it has no effect. In order to follow training process the algorithm exploits the property *Plots*. Setting the value *training-process* the algorithm display a figure which shows mini-batch loss and accuracy, validation loss and accuracy, and additional information on the training progress. Figure 45 is an example of the plot produced by the algorithm analysing only a small part of the dataset.

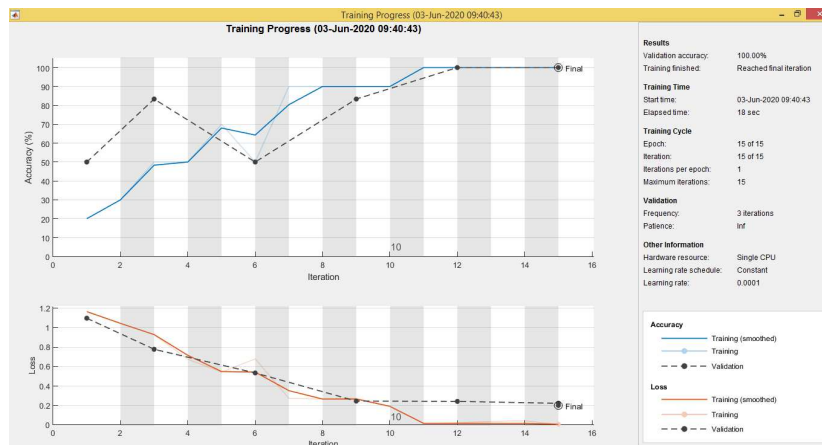


Figure 45. CNN training process.

Each plot in the figure shows three different curves: training (smoothed), training and validation. These lines can be very useful in order to follow the training process epoch by epoch and to detect possible over/under fitting. In particular validation curve is built starting from *ValidationData*. *ValidationData* is simply an image datastore containing all data utilized for validation during training, in this specific case 30% of the whole dataset. Each dot of the validation dashed line represent the evaluation of validation metrics and the iterations between one of these and the following one can be modified through the parameter *ValidationFrequency*. The *ValidationPatience* value is the number of times that the loss on the validation set can be larger than or equal to the previously smallest loss before network training stops. Setting this value to Inf means that the training process will not stop until the end of the last epoch.

Last two properties it is worth mentioning are: *Shuffle*, *every-epoch* and *SequenceLength*, *longest*. The first one permits the algorithm to shuffle the training data before each training epoch, and shuffle the validation data before each network validation. This property is useful to avoid discarding the same data every epoch. The second one lets the software pads the sequences so that all the sequences in a mini-batch have the same length as the longest sequence in the mini-batch. This option does not discard any data, though padding can introduce noise to the network.

All the other options are not relevant for this project and they are either set as default value or not utilized. For further information about this kind of training options please refers to <https://it.mathworks.com/help/deeplearning/ref/nnet.cnn.trainingoptionssgdm.html>.

4.4.1. Data augmentation

Listing 3. Data augmentation

```

1 imageAugmenter = imageDataAugmenter( ...
2     'RandXReflection',1);
3 augimdsTrain = augmentedImageDatastore(inputSize(1:2) ,...
4     imdsTrain, 'DataAugmentation',imageAugmenter);

```

In order to easily exploit data augmentation, MATLAB offers a function whose purpose is to configure a set of preprocessing options for image augmentation, such as resizing, rotation, and reflection. As it is possible to notice in listing 3, line 2, the proposed algorithm augments data using *RandXReflection*. If this property is activate then each image is reflected horizontally (figure 46) with 50% probability. The function *augmentedImageDatastore* (line 3) includes the images produced by the data augmentation process into the training set.

The reason why choosing horizontal reflection is that the result produced by this process are realistic fundus images. Moreover, if the training process is supported by this strategy then the final accuracy increase with respect to a classifier which do not exploit data augmentation.

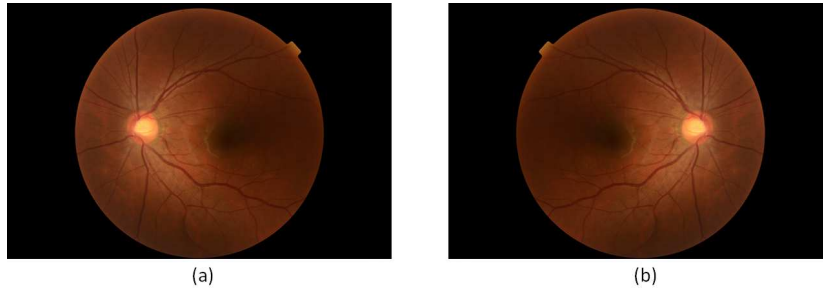


Figure 46. Example of data augmentation. Figure (a): original image. Figure (b): reflected image.

4.5. CNN testing

The following part of the algorithm regards the testing phase. After the training of each network a test is necessary in order to understand which performances the classifier can reach and to iteratively refine hyperparameters to achieve better and better results.

For this reason a specific dataset is built for CNN testing, usually this dataset is called testing set. Specifically these images are set aside during the training phase, once the network is ready to classify new images then the CNN picks from the testing set and produces a label for each fundus image (cataractous or not). Since all the images belonging to this dataset are labelled, the comparison between the true labels (the ones established by physicians or eye-specialist) and the predicted labels (the ones that result as output from the trained network) allows to estimate several performance parameters for example accuracy, sensitivity, specificity and others. A widely used tool that summarize all these parameter is the confusion matrix. MATLAB function *plotconfusion* provides this matrix directly from true and predicted labels. Confusion matrix and statistical measures of the performance will be described more in detail in the following chapter.

5. Results

Figure 47 shows one of the MATLAB confusion matrix obtained with the CNNs. In particular this confusion matrix is related to testing set 2.2 with *adaphisteq* as pre-processing. All the others confusion matrix produced are reported in appendix 1.

On this confusion matrix plot, the rows correspond to the predicted class (Output Class) and the columns correspond to the true class (Target Class). The diagonal cells (in green) correspond to images that are correctly classified (with cataract or not). The off-diagonal cells (in red) correspond to incorrectly classified fundus images. Both the number of observations and the percentage of the total number of observations are shown in each cell.

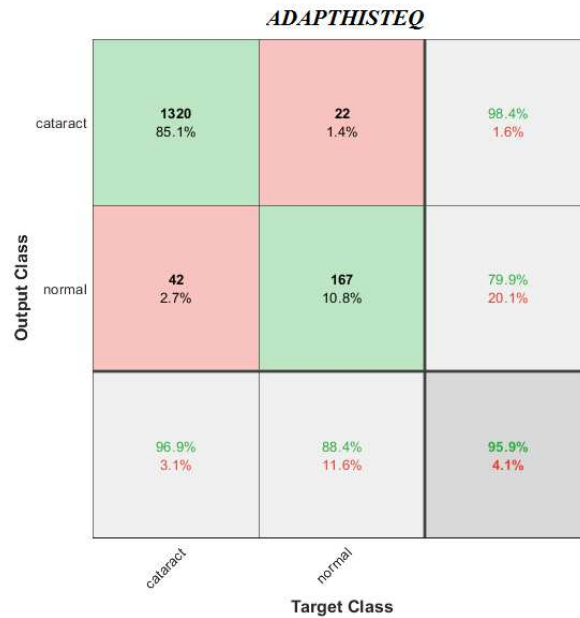


Figure 47. Confusion matrix produced by the MATLAB function *plotconfusion* that summarizes all the important statistics to evaluate CNN performances. It refers to the testing set 2.2 when fundus images are preprocessed with using contrast-limited histogram equalization (CLAHE).

In diagnostic medical tests (but not only) usually there is a specific terminology to define values expressed within a confusion matrix. Note that the terms "positive" and "negative" do not refer to the value of the condition of interest, but to its presence or absence. The condition itself could be a disease, so this work "positive" means "affected by cataract", while "negative" means "not affected by cataract".

- **True Positive (TP):** outcomes where the model correctly predicts the positive class. In this case 1320 out of 1362 real positive.

- **True Negative (TN)**: outcomes where the model correctly predicts the negative class. In this case 167 out of 189 real negative.
- **False Positive (FP)**: outcomes where the model incorrectly predicts the positive class. In this case 22 out of 1551 images.
- **False Negative (FN)**: outcomes where the model incorrectly predicts the negative class. In this case 42 out of 1551 images.

These values allow to compute several statistical measures. Formulas and values are reported below.

The column on the far right of the plot shows the percentages that refers to the following metrics. Starting from the top to the bottom:

- Precision or positive predictive value (PPV): $\frac{TP}{TP+FP} = 1 - FDR = 98.4\%$
- False discovery rate (FDR): $\frac{FP}{FP+TP} = 1 - PPV = 1.6\%$
- Negative predicted rate (NPV): $\frac{TN}{TN+FN} = 1 - FOR = 79.9\%$
- False omission rate (FOR): $\frac{FN}{FN+TN} = 1 - NPV = 20.1\%$

The row at the bottom of the plot shows the percentages that refers to the following metrics. Starting from the left to the right (first green percentages then red ones):

- Sensitivity or true positive rate (TPR): $\frac{TP}{TP+FN} = 1 - FNR = 96.9\%$
- Miss rate or false negative rate (FNR): $\frac{FN}{FN+TP} = 1 - TPR = 3.1\%$
- Specificity or true negative rate (TNR): $\frac{TN}{TN+FP} = 1 - FPR = 88.4\%$
- False positive rate (FPR): $\frac{FP}{FP+TN} = 1 - TNR = 11.6\%$

Finally the cell in the bottom right of the plot shows (in green) the overall accuracy: $\frac{TP+TN}{TP+TN+FP+FN} = 95.9\%$.

Although all these values have their statistical importance, for this project accuracy, sensitivity and specificity are chosen for evaluating and comparing CNN performances.

Accuracy measures how many samples are correctly labelled by the network with respect to all the classification performed. It is a general metric which could indicate the quality of the classifier.

Sensitivity measures how often a test correctly generates a positive result for people who have the condition that is being tested for. A test that is highly sensitive will flag almost everyone who has the disease and not generate many false-negative results.

Specificity measures a test's ability to correctly generate a negative result for people who do not have the condition that is being tested for. A high-specificity test will correctly rule out almost everyone who does not have the disease and will not generate many false-positive results.

It is important to recognize that sensitivity and specificity exist in a state of balance. Increased sensitivity usually comes at the expense of reduced specificity (meaning more false-positives). Likewise, high specificity usually means that the test has lower sensitivity (more false-negatives).

5.1. Dataset 1

This paragraph refers to the dataset presented in 4.2.1, which contains 393 images acquired by patients with cataract and 7007 images acquired by healthy subjects. As already mentioned, these images are acquired by fundus cameras and not from D-EYE. However dataset 1 results are useful because they represent a solid starting point for tuning hyperparameters and allow to compare the obtained results with those reported in literature and with those obtained by dataset 2.

Table 2 summarizes accuracy, sensitivity and specificity of the CNNs trained on dataset 1. The first row regards images without pre-processing while the second one fundus images after performing Fourier transform. Since a specific testing set was not considered necessary, these percentages refers to the validation set.

	Accuracy	Sensitivity	Specificity
Original Images	85.0%	90.7%	84.7%
FT	89.1%	89.8%	89.1%

Table 2. CNN performances related to dataset 1.

5.2. Dataset 2

This paragraph refers to the datasets presented in 4.2.2, which contains 3735 cataractous fundus images and 8100 healthy images. Once tuned hyperparameters thanks to dataset 1, it was possible to start training CNNs on dataset 2. The following tables present the most significant results. It is worth remember that dataset 2.1 refers to a validation set while datasets 2.2 and 2.3 are testing sets.

• Accuracy

	Original	FT	Green channel	<i>histeq</i>	<i>adapthisteq</i>
Dataset 2.1	90.3%	92.2%	91.0%	82.0%	86.1%
Dataset 2.2	79.7%	79%	80.5%	76.9%	81.0%
Dataset 2.3	87.0%	75.1%	86.2%	86.1%	89.6%

Table 3. CNN accuracy related to variations of dataset 2 with different pre-processing.

• Sensitivity

	Original	FT	Green channel	<i>histeq</i>	<i>adapthisteq</i>
Dataset 2.1	81.1%	66.8%	81.9%	77.0%	83.8%
Dataset 2.2	44.8%	32.2%	48.2%	56.5%	54.2%
Dataset 2.3	85.5%	72.3%	85.7%	87.0%	90.2%

Table 4. CNN sensitivity related to variations of dataset 2 with different pre-processing.

- **Specificity**

	Original	FT	Green channel	<i>histeq</i>	<i>adaphisteq</i>
Dataset 2.1	92.4%	97.8%	93.0%	83.1%	86.6%
Dataset 2.2	96.5%	99.0%	96.1%	86.6%	93.9%
Dataset 2.3	97.9%	95.2%	89.9%	79.9%	85.2%

Table 5. CNN specificity related to variations of dataset 2 with different pre-processing.



Figure 48. Histogram of the accuracy varying dataset and pre-processing.

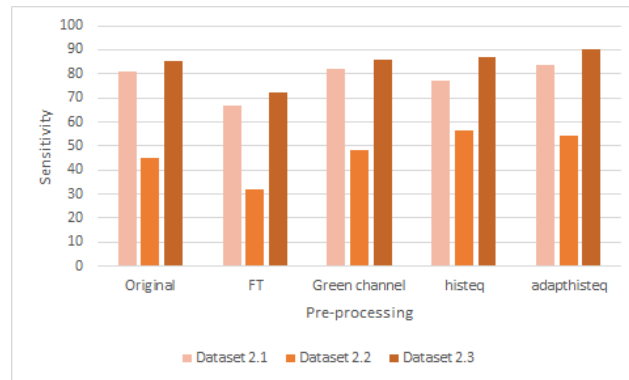


Figure 49. Histogram of the sensitivity varying dataset and pre-processing.

These statistic outcomes are encouraging. Looking at table 3 and figure 48, it is possible to notice that the CNN reaches a good accuracy on validation set but also on testing sets.

Focusing now on general performances, including also sensitivity and specificity. Some considerations can be done comparing dataset 1 and dataset 2. Looking at the data, the network is able to classify both fundus camera images and D-EYE images with comparable performances. In order to understand the importance of this result it is necessary to briefly recall the background of these two devices. On one hand there are fundus cameras, big devices designed to

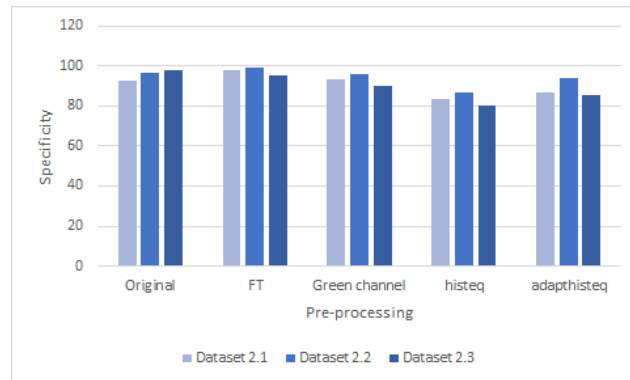


Figure 50. Histogram of the specificity varying dataset and pre-processing.

capture high quality fundus images and to perform a wide range of specific eye medical tests. This justifies why fundus cameras cost some thousands of US dollars. On the other hand there is D-EYE, a portable device whose cost is around few hundreds of US dollars. As already introduced, the main purpose of D-EYE is a massive screening of various eye pathologies, especially suitable in poor areas. Moreover fundus cameras can visualize a wider portion of the retina with respect to D-EYE. For all these reasons different output images between these two devices are unavoidable. Since the performance of a CNN is partially related to the quality of the images there could be a gap between the results obtained with dataset 1 with respect to the ones related to dataset 2. The comparability of the percentages indicates that the CNN specific for D-EYE images is well-built and well-trained since it can reach good accuracy (more than 90%), despite the quality difference of the images.

Regarding D-EYE images, tables 3-5 and figures 48-50 report the findings comparing different kinds of pre-processing. The function *adapthisteq* results to be the best pre-processing for this particular classification task. This is mainly due to the fact that D-EYE, as a screening medical device, requires a classification algorithm with an high sensitivity. Table 4 and figure 49 reveal that the CNN reaches the best sensitivity if classifies images pre-processed with *adapthisteq*. Moreover, considering the testing sets, this pre-processing method has the highest accuracy in terms of classification (table 3, figure 48).

5.3. Post-processing

A more detailed analysis of the results reveals that it is possible to exploit in order to further improve final accuracy. Describing DEYE-like tool, paragraph 4.2.2 reveals that from each fundus image the algorithm produces artificial images similar to the ones captured by D-EYE. Since all these images blocks comes form a single fundus image and since the effect of the cataract appears in each image, their diagnose must be the same. This assumption allows the use of majority voting technique. This is a well-known technique in literature, for example [60] and [70] increase their method performances thanks to majority voting.

This simple, but effective, strategy consists in finding the majority of a sequence of elements using linear time and constant space and then replacing all these elements with the majority value. The majority value, if there is one, is the element that occurs repeatedly for more than half of the elements of the input. In this case the elements are the labels given to each image after the classification process ("cataract" and "not cataract"). The algorithm iterates this process for each group of fundus images generated from the same input image in order to give a unique diagnosis.

This procedure is implemented after the classification process and the algorithm obtains a relevant performance improvement. Table 6 and figure 51-53 show the comparison between each statistical metrics before and after applying post-processing.

	Original	FT	Green channel	<i>histeq</i>	<i>adapthisteq</i>
Accuracy pre	87.0%	75.1%	86.2%	86.1%	89.6%
Accuracy post	92.1%	85.3%	93.2%	89.2%	95.9%
Sensitivity pre	85.5%	72.3%	85.7%	87.0%	90.2%
Sensitivity post	91.0%	83.3%	93.8%	90.7%	96.9%
Specificity pre	97.9%	95.2%	89.9%	79.9%	85.2%
Specificity post	100.0%	99.5%	88.9%	77.8%	88.4%

Table 6. Statistical metrics comparison before and after post-processing for each pre-processing applied .

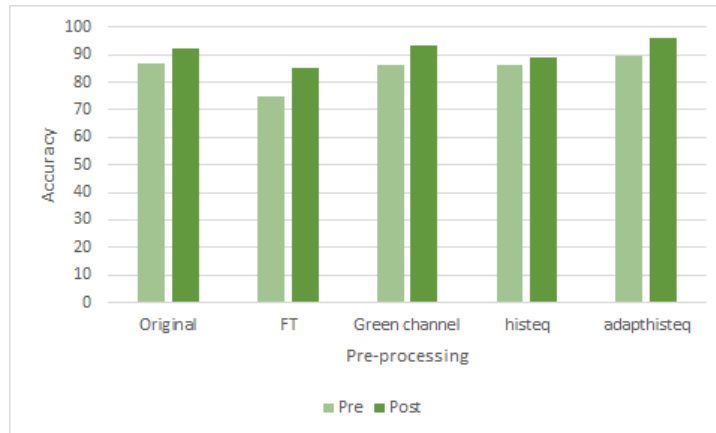


Figure 51. Histogram of the accuracy before and after post-processing for each pre-processing applied.

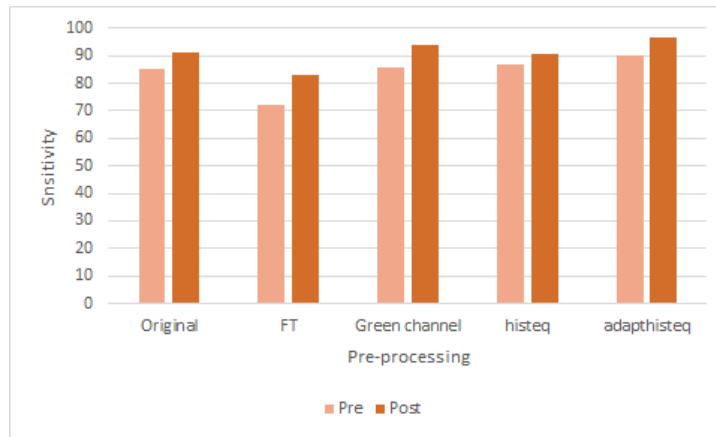


Figure 52. Histogram of the sensitivity before and after post-processing for each pre-processing applied.

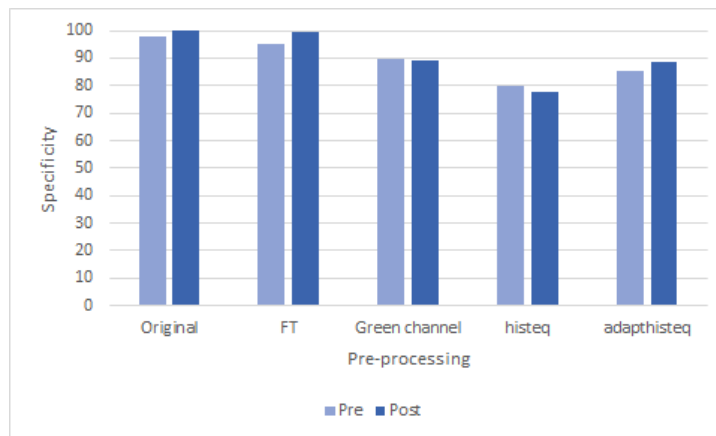


Figure 53. Histogram of the specificity before and after post-processing for each pre-processing applied.

As it is possible to notice, in general, all performances improve independently of any pre-processing methods used. Moreover *adapthisteq* keeps the best results in according to what has been concluded before. It obtains 95.9% accuracy, 96.9% sensitivity and 88.4% specificity.

It is worth mentioning that majority voting is a useful technique not only for the testing phase but it can represent the real use of an application for cataract detection, since from a video acquired with D-EYE various frames are selected and analysed. Therefore the final diagnosis could be the result a majority voting process that involves several frames extrapolated from the same video.

6. Conclusion

The World Health Organization affirms that cataract is responsible for 51% of world blindness. As people in the world live longer, the number of people with cataract is anticipated to grow. This terrible issue is an important cause of low vision in both developed and developing countries. D-EYE pursues a sight-saving mission aimed to achieve the vision of the homonymous company: to provide access to eye health everywhere [74]. The device, defined as "a pocket-sized ophthalmoscope that goes where patients are", could represent a breakthrough in rural areas healthcare also thanks to his close relationship with telemedicine.

In this thesis project, an algorithm able to automatically detect the presence of cataract analysing D-EYE fundus images has been developed. The core of the proposed algorithm is a custom convolutional neural network, which uses 20-layer to provide a binary classification (cataract vs not-cataract) from an image. Among the results obtained, contrast-limited adaptive histogram equalization (CLAHE), applied to green-channel images, turns out to be the best pre-processing method for this kind of fundus images while majority voting is a valid post-processing to improve final classification performances both in testing phase and in real applications. The proposed convolutional neural network together with pre- and post-processing is able to reach 95.9% of accuracy. Since D-EYE is mainly designed to be a screening device, sensitivity is an important statistical parameter in order to evaluate the classification effectiveness. An iteratively refining process regarding hyperparameters and the implementation of different strategies, either aimed at enhancing images quality or at improving algorithm performances, allowed the CNN to achieve 96.9% sensitivity in testing phase. The percentages just reported are comparable with the state-of-art deep learning algorithms for cataract detection.

These findings are encouraging and once again they demonstrate the power and the decisive role CNNs play in medical image processing. In our opinion deep learning will take hold more and more in future medical researches.

It is worth mentioning that D-EYE equipped with automatic detection algorithms can be useful not only for ophthalmologists but also for not-specialized medical personnel to perform examinations in geographically remote areas where an early eye disease detection could be crucial for blindness prevention. Moreover, thanks to his user-friendliness, students and professors can exploit D-EYE for education activities and also eye specialists can easily share outcomes and examination videos.

Currently cataract is not the only eye disease D-EYE can automatically detect. Indeed previous works, exploiting deep learning techniques, allowed the device to be able to estimate cup-to-disc ratio, a key parameter to evaluate the presence of glaucoma [75]. In the future, the company will develop other algorithms in order to detect a wider spectrum of diseases. For example, starting from the same basis of this code, a convolutional neural network could classify diabetic retinopathy, papilledema and other eye complaints.

In conclusion this work can be useful to help D-EYE to become more relevant and effective on the scene of ophthalmology and eye healthcare.

References

- [1] U Müller-Breitenkamp, C Ohrloff, and O Hockwin. Aspects of physiology, pathology and epidemiology of cataract. *Der Ophthalmologe: Zeitschrift der Deutschen Ophthalmologischen Gesellschaft*, 89(4):257–267, 1992.
- [2] Hassan Hashemi, Reza Pakzad, Abbasali Yekta, Mohamadreza Aghamirsalim, Mojgan Pakbin, Shahroukh Ramin, and Mehdi Khabazkhoob. Global and regional prevalence of age-related cataract: a comprehensive systematic review and meta-analysis. *Eye*, pages 1–14, 2020.
- [3] Barry D Kels, Andrzej Grzybowski, and Jane M Grant-Kels. Human ocular anatomy. *Clinics in dermatology*, 33(2):140–146, 2015.
- [4] Helga Kolb. Simple anatomy of the retina. *Webvision: The organization of the retina and visual system*, pages 13–36, 1995.
- [5] Segewkal H Heruye, Maffofou Nkenyi, N Leonce, Neetu U Singh, Dariush Yalzadeh, Kalu K Ngele, Ya-Fatou Njie-Mbye, Sunny E Ohia, and Catherine A Opere. Current trends in the pharmacotherapy of cataracts. *Pharmaceuticals*, 13(1):15, 2020.
- [6] David Allen and Abhay Vasavada. Cataract and surgery for cataract. *Bmj*, 333(7559):128–132, 2006.
- [7] Louis Pizzarello, Adenike Abiose, Timothy Ffytche, Rainaldo Duerksen, R Thulasiraj, Hugh Taylor, Hannah Faal, Gullapali Rao, Ivo Kocur, and Serge Resnikoff. Vision 2020: The right to sight: a global initiative to eliminate avoidable blindness. *Archives of ophthalmology*, 122(4):615–620, 2004.
- [8] Jason Singh, Sami Kabbara, Mandi Conway, Gholam Peyman, and Robin D Ross. Innovative diagnostic tools for ophthalmology in low-income countries. In *Novel Diagnostic Methods in Ophthalmology*. IntechOpen, 2019.
- [9] Carl Zeiss Meditec AG. Eye examination with the slit lamp, 2001.
- [10] Simon N Madge. *Clinical techniques in ophthalmology*. Elsevier Health Sciences, 2006.
- [11] Nishtha Panwar, Philemon Huang, Jiaying Lee, Pearse A Keane, Tjin Swee Chuan, Ashutosh Richhariya, Stephen Teoh, Tock Han Lim, and Rupesh Agrawal. Fundus photography in the 21st century—a review of recent technological advances and their implications for worldwide healthcare. *Telemedicine and e-Health*, 22(3):198–208, 2016.
- [12] Chen Ma, Dewen Cheng, Chen Xu, and Yongtian Wang. Design, simulation and experimental analysis of an anti-stray-light illumination system of fundus camera. In *Optical Design and Testing VI*, volume 9272, page 92720H. International Society for Optics and Photonics, 2014.

-
- [13] Drew Dickson, Samiksha Fouzdar-Jain, Collin MacDonald, Helen Song, Daniel Agraz, Linda Morgan, and Donny Suh. Comparison study of fundoscopic exam of pediatric patients using the d-eye method and conventional indirect ophthalmoscopic methods. *Open Journal of Ophthalmology*, 7(3):145–152, 2017.
- [14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.
- [15] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [16] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.
- [17] Enzo Grossi and Massimo Buscema. Introduction to artificial neural networks. *European journal of gastroenterology & hepatology*, 19(12):1046–1054, 2007.
- [18] Erik G Learned-Miller. Introduction to supervised learning. *I: Department of Computer Science, University of Massachusetts*, 2014.
- [19] Michael A Nielsen. *Neural networks and deep learning*, volume 2018. Determination press San Francisco, CA, USA:, 2015.
- [20] Willow Garage. Personal robot 2 (pr2). *Available: www.willowgarage.com*, 2013.
- [21] Jianxin Wu. Introduction to convolutional neural networks. *National Key Lab for Novel Software Technology. Nanjing University. China*, 5:23, 2017.
- [22] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.
- [23] Keiron O’Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.
- [24] MathWorks. Introducing deep learning with matlab. *Ebook: <https://it.mathworks.com/campaigns/offers/deep-learning-with-matlab.html>*.
- [25] Xingye Qiao and Yufeng Liu. Adaptive weighted learning for unbalanced multiclass classification. *Biometrics*, 65(1):159–168, 2009.
- [26] Wil MP Van der Aalst, Vladimir Rubín, HMW Verbeek, Boudewijn F van Dongen, Ekkart Kindler, and Christian W Günther. Process mining: a two-step approach to balance between underfitting and overfitting. *Software & Systems Modeling*, 9(1):87, 2010.
- [27] Lutz Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 1998.

-
- [28] Rich Caruana, Steve Lawrence, and C Lee Giles. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pages 402–408, 2001.
- [29] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.
- [30] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.
- [31] <http://www.image-net.org/>.
- [32] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.
- [33] Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.
- [34] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.
- [35] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.
- [36] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.
- [37] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th annual international conference on machine learning*, pages 689–696, 2009.
- [38] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.
- [39] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.
- [40] Kenji Suzuki, Isao Horiba, Kazuo Ikegaya, and Michio Nanki. Recognition of coronary arterial stenosis using neural network on dsa system. *Systems and Computers in Japan*, 26(8):66–74, 1995.
- [41] Kenji Suzuki, Isao Horiba, and Noboru Sugie. Edge detection from noisy images using a neural edge detector. In *Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop (Cat. No. 00TH8501)*, volume 2, pages 487–496. IEEE, 2000.

-
- [42] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.
- [43] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [44] Kenji Suzuki. Overview of deep learning in medical imaging. *Radiological physics and technology*, 10(3):257–273, 2017.
- [45] Andreas Maier, Christopher Syben, Tobias Lasser, and Christian Riess. A gentle introduction to deep learning in medical image processing. *Zeitschrift für Medizinische Physik*, 29(2):86–101, 2019.
- [46] Yu-Cheng Chen, Derek Jin-Ki Hong, Chia-Wei Wu, Muralidhar Muppapurapu, et al. The use of deep convolutional neural networks in biomedical imaging: A review. *Journal of Orofacial Sciences*, 11(1):3, 2019.
- [47] Fang Liu, Zhaoye Zhou, Hyungseok Jang, Alexey Samsonov, Gengyan Zhao, and Richard Kijowski. Deep convolutional neural network and 3d deformable approach for tissue segmentation in musculoskeletal magnetic resonance imaging. *Magnetic resonance in medicine*, 79(4):2379–2391, 2018.
- [48] Jinlian Ma, Fa Wu, Jiang Zhu, Dong Xu, and Dexing Kong. A pre-trained convolutional neural network based method for thyroid nodule diagnosis. *Ultrasonics*, 73:221–230, 2017.
- [49] Mathias Unberath, Jan-Nico Zaech, Sing Chun Lee, Bastian Bier, Javad Fotouhi, Mehran Armand, and Nassir Navab. Deepdrr—a catalyst for machine learning in fluoroscopy-guided procedures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 98–106. Springer, 2018.
- [50] Bo Zhu, Jeremiah Z Liu, Stephen F Cauley, Bruce R Rosen, and Matthew S Rosen. Image reconstruction by domain-transform manifold learning. *Nature*, 555(7697):487–492, 2018.
- [51] Daniel SW Ting, Lily Peng, Avinash V Varadarajan, Pearse A Keane, Phil Burlina, Michael F Chiang, Leopold Schmetterer, Louis R Pasquale, Neil M Bressler, Dale R Webster, et al. Deep learning in ophthalmology: the technical and clinical considerations. *Progress in retinal and eye research*, 2019.
- [52] Parampal S Grewal, Faraz Oloumi, Uriel Rubin, and Matthew TS Tennant. Deep learning in ophthalmology: a review. *Canadian Journal of Ophthalmology*, 53(4):309–313, 2018.
- [53] Daniel Shu Wei Ting, Louis R Pasquale, Lily Peng, John Peter Campbell, Aaron Y Lee, Rajiv Raman, Gavin Siew Wei Tan, Leopold Schmetterer, Pearse A Keane, and Tien Yin Wong. Artificial intelligence and deep learning in ophthalmology. *British Journal of Ophthalmology*, 103(2):167–175, 2019.

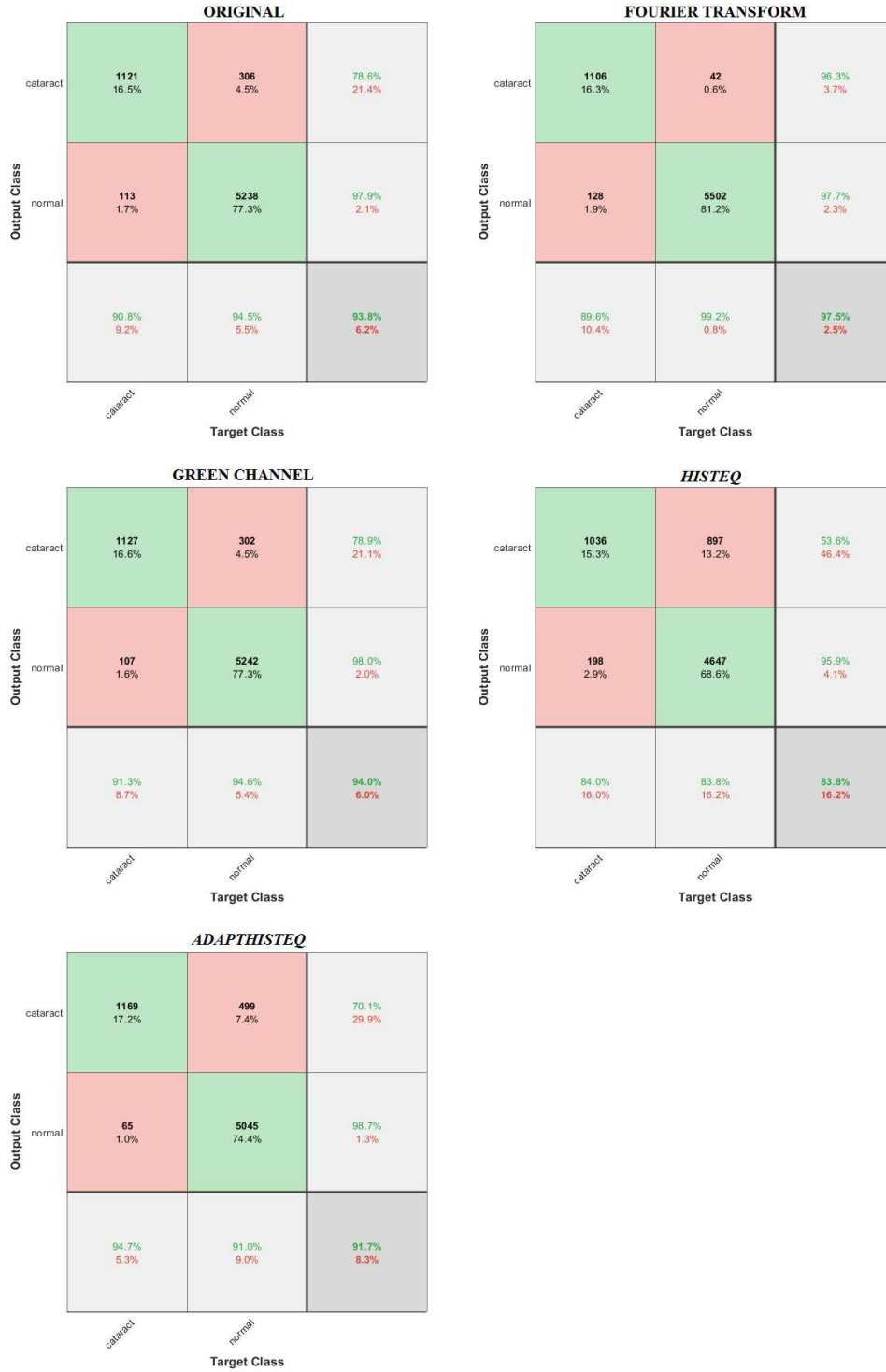
-
- [54] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, 316(22):2402–2410, 2016.
- [55] Cecilia S Lee, Doug M Baughman, and Aaron Y Lee. Deep learning is effective for classifying normal versus age-related macular degeneration oct images. *Ophthalmology Retina*, 1(4):322–327, 2017.
- [56] Florian Schiffers, Zekuan Yu, Steve Arguin, Andreas Maier, and Qiushi Ren. Synthetic fundus fluorescein angiography using deep neural networks. In *Bildverarbeitung für die Medizin 2018*, pages 234–238. Springer, 2018.
- [57] Anmar M Abdul-Rahman, Tim Molteno, and Anthony CB Molteno. Fourier analysis of digital retinal images in estimation of cataract severity. *Clinical & experimental ophthalmology*, 36(7):637–645, 2008.
- [58] Jin Zheng, Liye Guo, Lihui Peng, Jianqiang Li, Jijiang Yang, and Qingfeng Liang. Fundus image based cataract classification. In *2014 IEEE International Conference on Imaging Systems and Techniques (IST) Proceedings*, pages 90–94. IEEE, 2014.
- [59] Liye Guo, Ji-Jiang Yang, Lihui Peng, Jianqiang Li, and Qingfeng Liang. A computer-aided healthcare system for cataract classification and grading based on fundus image analysis. *Computers in Industry*, 69:72–80, 2015.
- [60] Ji-Jiang Yang, Jianqiang Li, Ruifang Shen, Yang Zeng, Jian He, Jing Bi, Yong Li, Qinyan Zhang, Lihui Peng, and Qing Wang. Exploiting ensemble learning for automatic cataract detection and grading. *Computer methods and programs in biomedicine*, 124:45–57, 2016.
- [61] Sucheta Kolhe and Shanthi K Guru. Remote automated cataract detection system based on fundus images. *International Journal of Innovative Research in Science, Engineering and Technology*, 5(6):23, 2016.
- [62] V Harini and V Bhanumathi. Automatic cataract classification system. In *2016 International Conference on Communication and Signal Processing (ICCSP)*, pages 0815–0819. IEEE, 2016.
- [63] Zhiqiang Qiao, Qinyan Zhang, Yanyan Dong, and Ji-Jiang Yang. Application of svm based on genetic algorithm in classification of cataract fundus images. In *2017 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–5. IEEE, 2017.
- [64] Wenai Song, Ping Wang, Xudong Zhang, and Qing Wang. Semi-supervised learning based on cataract classification and grading. In *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, volume 2, pages 641–646. IEEE, 2016.
- [65] Meimei Yang, Ji-Jiang Yang, Qinyan Zhang, Yu Niu, and Jianqiang Li. Classification of retinal image for automatic cataract detection. In *2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013)*, pages 674–679. IEEE, 2013.

-
- [66] Linglin Zhang, Jianqiang Li, He Han, Bo Liu, Jijiang Yang, Qing Wang, et al. Automatic cataract detection and grading using deep convolutional neural network. In *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, pages 60–65. IEEE, 2017.
- [67] Yanyan Dong, Qinyan Zhang, Zhiqiang Qiao, and Ji-Jiang Yang. Classification of cataract fundus image based on deep learning. In *2017 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–5. IEEE, 2017.
- [68] Jianqiang Li, Xi Xu, Yu Guan, Azhar Imran, Bo Liu, Li Zhang, Ji-Jiang Yang, Qing Wang, and Liyang Xie. Automatic cataract diagnosis by image-based interpretability. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3964–3969. IEEE, 2018.
- [69] Hongyan Zhang, Kai Niu, Yanmin Xiong, Weihua Yang, ZhiQiang He, and Hongxin Song. Automatic cataract grading methods based on deep learning. *Computer methods and programs in biomedicine*, 182:104978, 2019.
- [70] Xi Xu, Linglin Zhang, Jianqiang Li, Yu Guan, and Li Zhang. A hybrid global-local representation cnn model for automatic cataract grading. *IEEE journal of biomedical and health informatics*, 2019.
- [71] FA Hashim, NM Salem, and AF Seddik. Optic disc boundary detection from digital fundus images. *Journal of Medical Imaging and Health Informatics*, 5(1):50–56, 2015.
- [72] Tom McReynolds and David Blythe. *Advanced graphics programming using OpenGL*. Elsevier, 2005.
- [73] Etta D Pisano, Shuquan Zong, Bradley M Hemminger, Marla DeLuca, R Eugene Johnston, Keith Muller, M Patricia Braeuning, and Stephen M Pizer. Contrast limited adaptive histogram equalization image processing to improve the detection of simulated spiculations in dense mammograms. *Journal of Digital imaging*, 11(4):193, 1998.
- [74] <https://www.d-eyecare.com/>.
- [75] Fabio Scarpa, Alexa Berto, Alessia Colonna, and Alberto Scarpa. Development of an automated analysis for glaucoma screening of videos acquired with smartphone ophthalmoscope. *Investigative Ophthalmology & Visual Science*, 61(1), 2020.

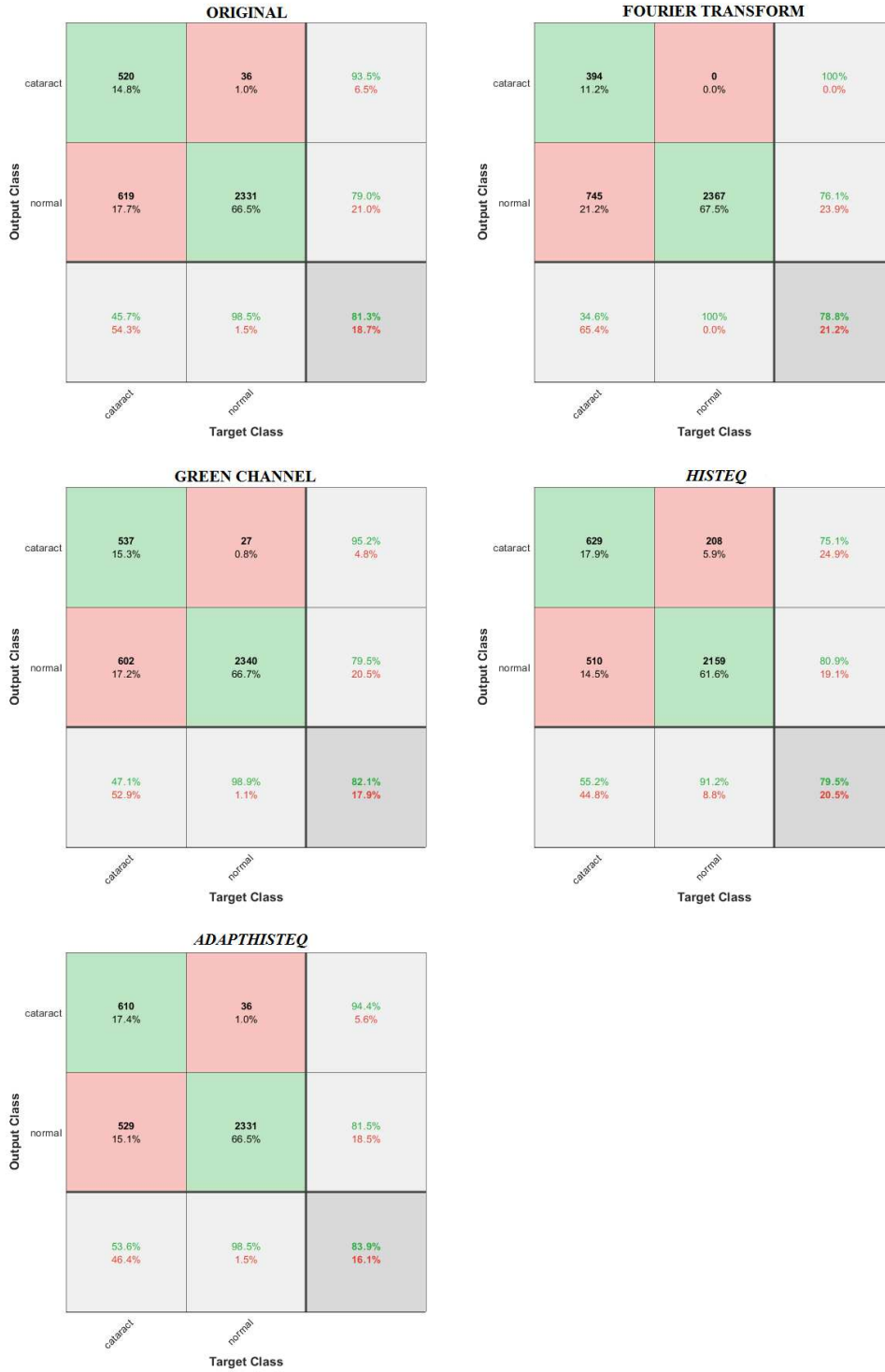
Appendix 1

This appendix contains confusion matrices produced by the algorithm. Each page refers to a different image dataset, specified as header. From left to right and from top to down figures show matrices related to no pre-processing, pre-processed with Fourier transform, with green channel, with *histeq*, with *adapthis-teq*, respectively.

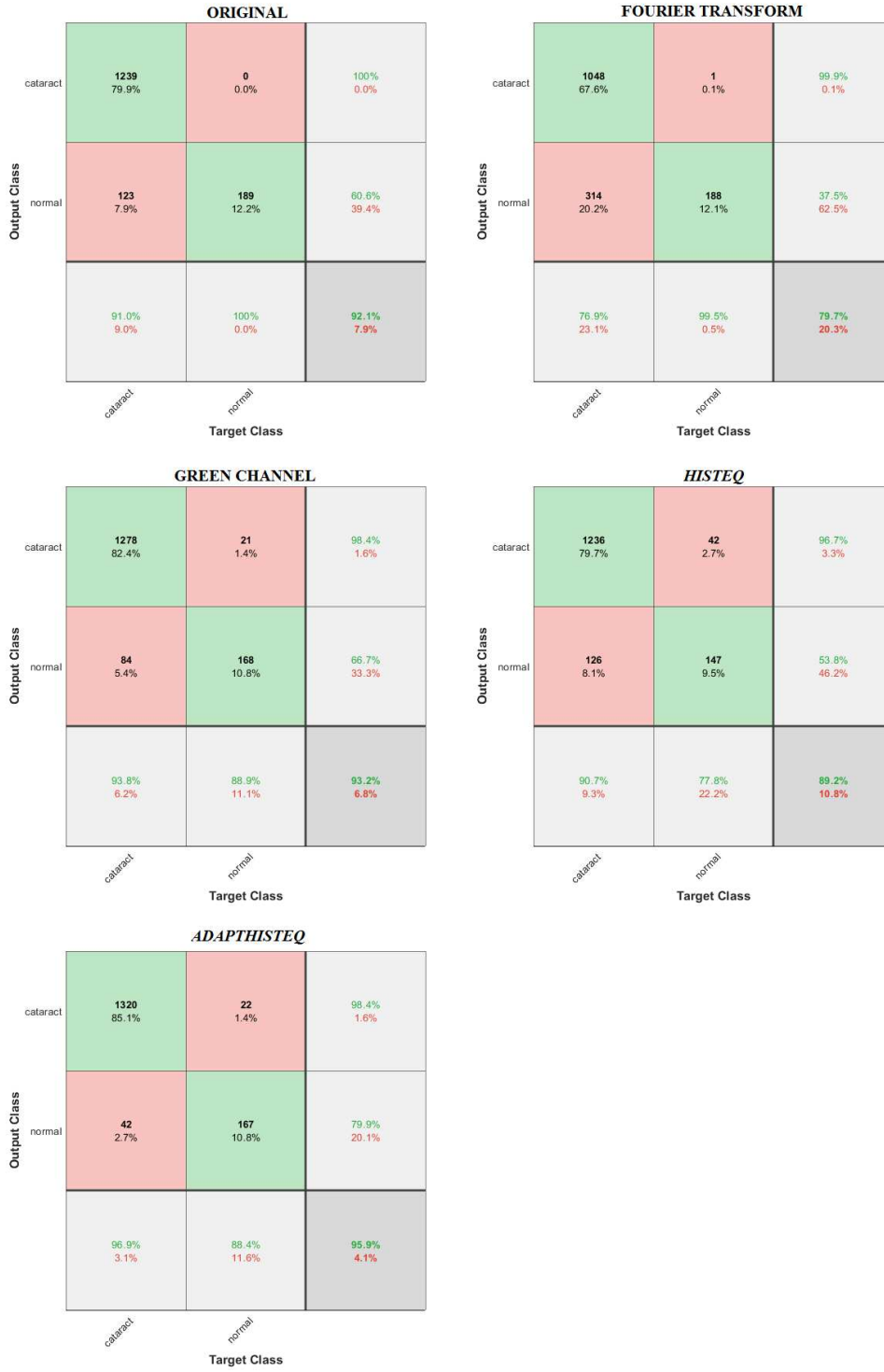
• Dataset 2.1



• Dataset 2.2



• Dataset 2.3



Acknowledgements

I would like to say thank you to Prof.ssa Maria Pia Saccomani for being my supervisor and for choosing to support me in my next experience.

Also I have to say thanks to Dott.ssa Alexa Berto and Ing. Fabio Scarpa. Despite the recent pandemic events, you were able to guide me along all this project pathway solving all my doubts promptly.

During my whole life I have never felt lonely. Thanks to my family to be there and to support me in every situation. Even if your way maybe it is not the usual way, well this made me stronger and made me who I am now. Thanks. Thanks mom for your infinite patience (this is not a commonplace for her). Thanks dad for sharing your knowledge with me since I was one (or even before). Thanks sister for the right thing at the right moment. You are growing up quickly. You can achieve whatever goals you want, maybe now you are following me but one day you will be in the front line.

A really special thanks to Rizzo's family. Raffaella, Davide, Diletta, Lapo and also Gio (as acquired member). Thanks for accepting me as I am (sometimes it is not so easy). When I am with you I feel I am in family.

The biggest thanks goes to You. You are the person I think of when Sum 41 play "With Me". It's true, I am nothing without you. Day by day you change my life so I can be better than the day before. We are growing together and we will grow together overcoming obstacles and sharing adventures. These few words are not enough (and you know I am not so good with words). I hope to be able to be your happiness as you are mine.

You raise me up to more than I can be.