Mémoire de Master

Mention Électronique
présenté par

KAABECHE Abderrazak Lakhder

&

ZITOUNE Mohamed Redha

Cells Detection of microscopic tests using YOLO V5

Proposé par :Mme Boughrira Hamida

Année Universitaire 2020-2021

Acknowledgement

Above all, we thank the almighty god for giving us the strength in order to carry out this work.

It is offered to us here, by these few lines, the possibility of thanking the people who contributed to this brief.

We would particularly like to thank our supervisor Madame Bougherira Hamida to know what she instilled in us, for his invaluable advice and directions, and especially for the support he gave us.

We would also like to thank who agreed to evaluate this work and to help improve it with their constructive comments.

And all the people who have contributed directly or indirectly to the culmination of this work

We express to you our gratitude and our deep respect, for all here we say "Thank you".

We also thank our families and friends, who helped us, supported and who offered us, to establish this work.

Dedication

I dedicate my dissertation work to my family and many friends. A special feeling of gratitude towards my parents, Abderrahmane and Safia whose words of encouragement and push for tenacity ring in my ears. My brother Fateh and sisters Ahlam, Ilhem, Ibtissem, Wadia and Inaam have never left my side and are very special.

I also dedicate this dissertation to many friends who have supported me throughout the process. I will always appreciate all they have done, especially Ishak Daoud for helping me advance when I get stuck and develop my skills, Fella Bouhafs for doc work and adjustements also for guidance.

I dedicate this work and give special thanks to my Homies Achraf Daoud, Abderrahmane Benmoffok, Samy Alliouet ,Moncef Bouguerra and Youcef Menacere for all the fun we had. And not to forget my buddies Shaima, Mecheri, Jasminka, nermine, Rachid, Raouf, Ikram , Marwan and all Among Us team.

At the end I'd like to dedicate this work to my neighbor Abdessalem Chebab aka Mustapha may his soul find peace and may the lord allow him to rest in his paradise Inchallah .

Johnny

Dédicace

Je dédie ce travail

A mon père et ma mère pour leur amour inestimable, leurs sacrifices, leur confiance, leur soutien et toutes les valeurs qu'ils ont su m'inculquer.

A ma grand-mère, pour toute l'affection qu'elle m'a témoignée et pour ses précieux encouragements.

A mes sœurs et mes frères pour leur tendresse, leur complicité et leur présence malgré la distance qui nous sépare.

A mes oncles et tantes de la famille ainsi que de la famille pour leurs mots d'encouragement et leur gentillesse.

A tous mes camarades et amis

Redha

ملخص: الهدف من هذا المشروع هو تبسيط وتسريع سير عمل الاختبارات المجهرية في المستشفيات الجزائرية والمختبرات المنخفضة بسبب انخفاض ميزانية المعدات ، حيث يعتمد على الكشف عن الخلايا باستخدام خوارزمية التعلم الكشف عن الأشياء في الوقت الفعلي ، استخدمنا 700 صورة مجهرية زائفة تم YOLO v5. العميق المعروفة باسم إنشاؤها والتي تم تضخيمها لاحقًا إلى 1600 صورة والتي أعطت نتائج مرضية عند اختبارها على صور مجهرية حقيقية.

كلمات المفاتيح: الاختبارات المجهرية. الكشف عن الخلايا. صورة مجهرية زائفة. التعلم العميق.

Résumé : L'objectif de ce projet est de simplifier et d'accélérer le flux de travail des tests microscopiques dans les hôpitaux algériens et les laboratoires bas de gamme en raison du faible budget d'équipement, il est basé sur la détection des cellules en utilisant un algorithme d'apprentissage profond connu sous le nom de YOLO v5, il est utilisé pour la classification et la détection d'objets en temps réel, nous avons utilisé 700 images pseudo microscopiques générées qui ont ensuite été augmentées à 1600 images qui ont donné des résultats satisfaisants lorsqu'elles ont été testées sur des microscopes réels.

Mots clés : Test microscopique;détection des cellules;apprentissage profond ;YOLO v5 ;détection d'objet ;classification ;images pseudo microscopiques .

Abstract : The objective of this project is to simplify and accelerate the workflow of microscopic tests in Algerian Hospitals and under budget laboratories, it's based on detecting cells using a Deep Learning algorithm known as YOLO v5 it is used for classification and real time object detection, we used a 700 generated pseudo microscopic images that was later augmented into 1600 images which gave a satisfying results when tested on real microscopic ones.

Keywords : Microscopic tests ;detecting cells ;Deep Learning;YOLO v5;object detection;classification; pseudo microscopic images.

Listes des acronymes et abréviations

AI:	Artificial Inteligence
AIDS:	Acquired ImmunoDeficiency Syndrome
ANN:	Artificial Neural Network
CBEU:	Cyto Bacteriological Exam of Urines
CFU:	Colony Forming Unit
CLL:	Chronic Lymphocytic Leukemia
CNN:	Convolutional Neural Network
COCO:	Common Objects in Context
CONV:	Convolution
CRP:	C-reactive protein
CSF:	Cerebrospinal fluid
CSP:	Cross Stage Partial
ENT:	Ear, Nose and Throat surgery
FPN:	Feature Pyramid Network
GPU:	Graphical Processing Unit
LDH:	lactate dehydrogenase
MLP:	Multilayer Perceptron
PAN:	Path Aggregation Network
RBC:	Red Blood Cells
RGB:	Red Green Blue
TPU:	Tensor Processing Unit
UEC:	Urea Electrolytes and Creatinine
UGC:	Urine Germ Count
VGG:	Visual Geometry Group
XOR:	Exclusively-OR
YAML:	Yet Another Markup Language
YOLO:	You Only Look Once

Table of contents

List of Figures

List of Tables:

General Introduction

There is always a need of qualified workers in the health section and that's always the case in our country Algeria exactly in laboratories where microscopic tests are performed, we aim to simplify and make their process less time consuming and easier.

Due to the low budget a public hospitals receives its laboratory can't afford high end machines that exist to perform automation of work which are very expensive even for private labs and only big labs in the country who have a high patients count would invest in such devices.

These microscopic tests consist of observing pathogens under light microscope and trying to recognize then count or approximate their appearances, this flow of work can be very exhausting because of the amount of concentration and the position of eyes on the scope plus memorization of which cells was counted etc.… which might lead to raising the error ratio which lowers the accuracy of the interpretation.

High end devices uses technologies such as projection of 3D images which then processed through filters and algorithm like Low pass, High pass, threshold, Watershed and then ends up with points of interest which are the cells and each type of cells will have its method.

New ways do involve image processing with simple equipment that runs on a variety of devices but yet to reach to professional use , with Deep learning a system can be trained to detect these pathogens just by an optical observation especially the one we will use YOLO which stands for You Only Look Once so just by first look it will try to determine the boundaries and the class of the pathogen.

This thesis is composed of 3 chapters:

The first chapter: we will dive into different pathogens and microscopic tests.

The second chapter: we will discuss different models and architectures that involve Deep learning.

The third chapter: is about generating pseudo microscopic images that later was used to train and the discussion of the results.

Chapter 1 Cells and microscopy tests

1.1 Introduction:

Microbiology focuses on the study of microorganisms, whether they are bacteria, fungi, protozoa or viruses. A deep understanding of their physiology, genetics, and interactions will help us understand the biological world on a microscopic scale.

However, some microorganisms are pathogenic, that is, they can cause diseases in humans (host), such as tuberculosis and plague (bacterial origin), malaria (protozoa) or AIDS (virus). Our knowledge allows us today to discover immunological methods for the treatment of certain diseases, such as cancer. However, disrupting these immune functions can lead to the destruction of healthy tissue, leading to diseases such as diabetes or lupus. In short, these two disciplines make it possible to use microorganisms to promote human well-being and to develop methods and treatments to regain health.

In this chapter we will learn about microscopic test and the pathogens we can observe that later we will develop a solution to recognize and automate the workflow of the microscopic test.

1.2 Pathogens:

1.2.1 Bacteria:

A bacterium is a microscopic organism generally formed by a single cell. This cell is distinguished from the cells of most other organisms, both unicellular and multicellular, by its simplified organization: bacterial cells are prokaryotic while those of other organisms are eukaryotic[1].

They range in size from 0.2 microns or less to 5.00 micron "giant" cells, but most are between 1 and 10 microns in size. They can most often be observed under an ordinary microscope (magnification of about 1000 times)[1].

1.2.1.1 Macroscopic examination of a biological sample:

Any bacterial infection is accompanied, in addition to the presence of bacteria, by biological signs related to inflammation with the possible presence of leukocytes, in particular polynuclear cells.

1.2.1.1.1 Microscopic examination:

Microscopic examination is of diagnostic interest above a certain threshold... So often, no macroscopic or visible abnormality will be noted, hence the need to look for bacteria and cellular elements (leukocyte) under the light microscope[2].

This examination is also useful to specify the nature of the macroscopic anomalies observed. The microscopic examination allows distinguishing different morphological types of bacteria:
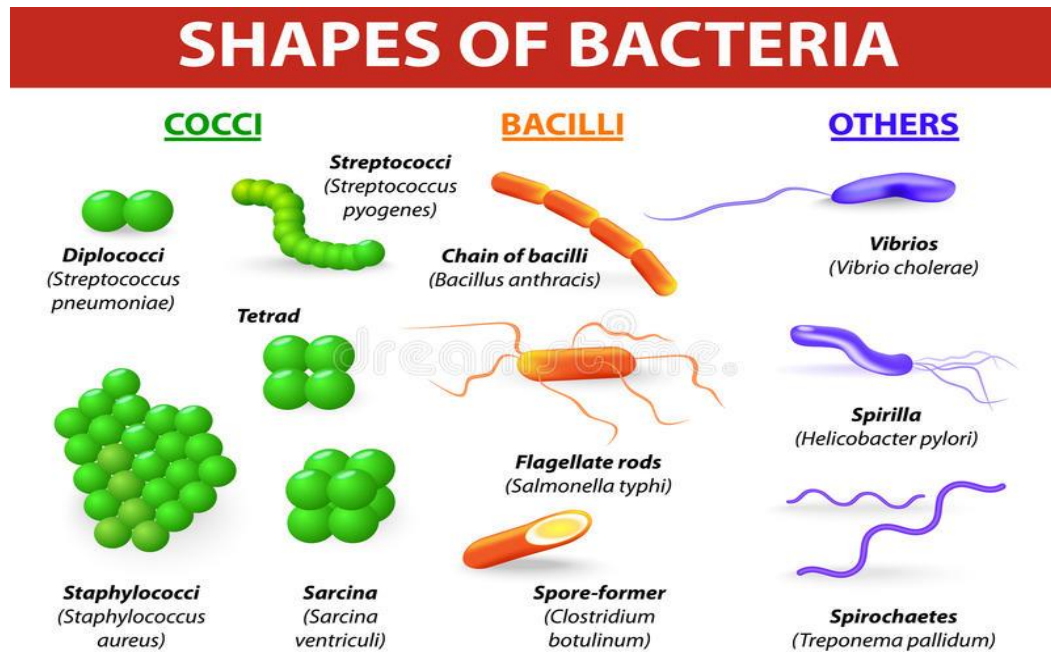
Figure 1.1: Spherical shape: The spherical shape characterizes the cocci.

These bacteria are subdivided into:

- Diplococcic: cells arranged in pairs and sometimes presenting a more or less accentuated tapered or kidney-shaped aspect.
- Streptococci: cells arranged in chains.
- Tetrads: grouping of 4 cells in a square on the same plane.
- Sarcines: regular cubes of 8 cells or multiples of 8.
- Staphylococci: cells gathered in irregular clusters.

Cylindrical form, in stick. Two types are distinguished:

- Bacilli: straight rods with rounded, square, tapered, spindle-shaped ends.
- Vibrios: crescent or sickle-shaped rods.
- Helical or spiral shape. These bacteria which have the appearance of spiral filaments can be differentiated by their length, the number and the amplitude of their undulations.
- Branched form. Some bacteria, such as actinomycetes, can have branches.
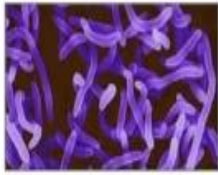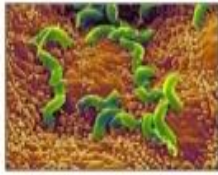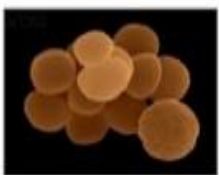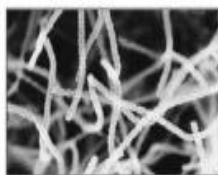
Circulaires (Coccus)	Bâtonnets (Bacilles)	Courbés	Autres formes
Diplo- (par paire)	Coccobacilles (ovales)	Vibrio (bâtonnets courbés)	Helicobacter (en hélice)
Strepto- (en chaîne)	Streptobacilles	Spirilles (spirale)	Corynebacterium (renflée)
Staphylo- (en amas)	Mycobactéries	Spirochète (spirale)	Streptomyces (filaments)

Figure 1.2: The diversity of bacteria

1.2.2 Yeasts:

They are unicellular fungi whose size varies from 2 to 20 µm. and which is therefore greater than that of bacteria, which allows them to be easily distinguished. Yeasts can be presented in several forms[3]:

- Round or oval shape: yeasts and spores.
- Pseudo filamentous form: chains of yeasts forming filaments (different species of Candida).
- Filamentous form: formation of true mycelium (C. albicans).

Figure 1.3: Yeast under light microscope in urine in urine sample

1.2.3 Viruses:

A virus is an infectious microscopic particle that can only replicate itself by entering a cell and using its cellular machinery. The viruses that infect bacteria are called bacteriophages. There are viruses that infect animals and others that infect plants. If they cause disease, viruses can be considered as pathogens[4].
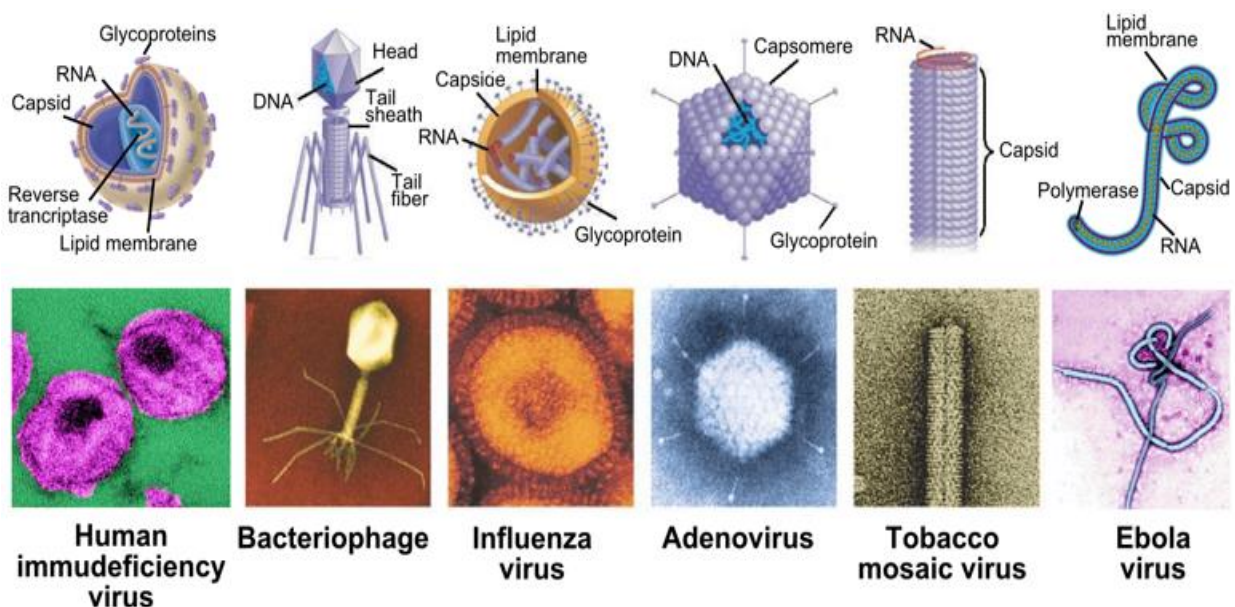
Figure 1.4: The different forms of some viruses

1.2.3.1 Pathological liquids to be tested in a medical laboratory:

Example:

Urine, cerebrospinal fluid and pleural fluid which may be cloudy or haematic.

Figure 1.5: Urine sample

1.2.4 Urines:

Normal urine is a pale yellow, clear, slightly acidic liquid that may contain very small amounts of red blood cells and leukocytes (less than 5,000/mL). It contains 95% water and 5% solutes: urea, creatinine, uric acid, ammonium ions, ketone bodies, electrolytes (sodium, potassium, calcium, etc.) Its composition and concentration vary according to the needs to ensure the homeostasis of the organism. Urine does not normally contain proteins, carbohydrates or lipids. Their presence in urine is a sign of pathology[5].

1.2.4.1 Cytobacteriological examination of urine (CBEU):

The CBEU is the most requested analysis in the medical microbiology laboratory.

It allows to:

- Highlight signs of inflammation of the urinary tract (expressed as leukocyturia)
- Identify, count and study the antibiotic sensitivity of the pathogen.

Interpretation of the UEC is based on:

- Cytological examination
- Leukocyturia
- Hematuria
- Epithelial cells
- Cylinder
- Crystals

- Bacteriological examination
- Bacteriuria
- Isolation
- Interpretation of results.

1.2.5 Leukocytes:

Also known as white blood cells are immune system cells that form in the bone marrow and are present in the blood, lymph, lymphoid organs and certain tissues. Since white blood cells increase in the event of infection, inflammation or bone marrow dysfunction, measuring them is very useful. It is also used to monitor the body's response to various treatments[6].

 There are three main categories of white blood cells:

1.2.5.1 Monocytes:

Monocytes (macrophages) represent 7% of white blood cells. In adults, there are 2 to 8 monocytes per 100 white blood cells. Like many blood cells, monocytes are produced by an organic tissue contained in certain bones, the bone marrow. They are the main actors of phagocytosis after their transformation into phagocytes[6].

1.2.5.2 Lymphocytes:

Lymphocytes represent 30% of leukocytes). A distinction is made between B lymphocytes (B lymphocytes produce antibodies specific to an antigen: the antibodies attach themselves to antigens, such as bacteria, in order to destroy them) and T lymphocytes (secreted by the thymus before transforming into thymocytes, these cells divide and multiply rapidly when they are exposed to an antigen). They are part of acquired immunity: that is, they are not present at birth or immediately, unlike the innate immune response, and only come into play when a person's immune system encounters and recognizes foreign infectious agents called "antigens" (bacteria, viruses, fungi, parasites...)[6].

Under light microscopy, they are small cells, about 7 µm in diameter with a nucleus occupying almost the entire cell.

1.2.5.3 Polynuclear cells (granulocytes):

Polynuclear cells carry out what is called "phagocytosis", i.e. they attack foreign bodies in case of infection. They are part of the innate immunity system, present from birth. It is the first response of the organism when attacked by a pathogen. Under light microscopy,

these cells are about 12 μm in diameter, the nucleus is generally trilobed but the number of lobes varies from 2 to 5[6].

There are three types of polynuclear cells:

- ✓ Neutrophils (60% of leukocytes).
- ✓ Eosinophils (2 to 4% of leukocytes).
- ✓ Basophils (1% of leukocytes).
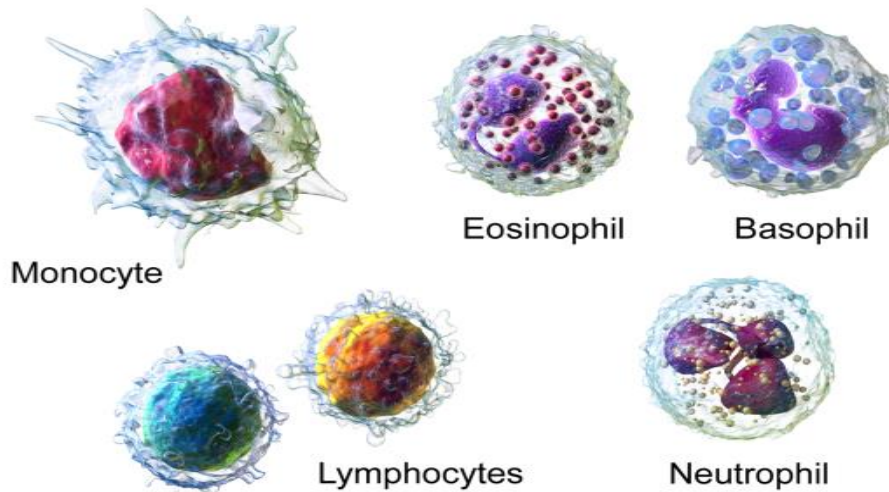
Figure 1.6: Different types of white blood cells

Leukocyturia> 104/mL indicates an inflammatory reaction following a urinary tract infection.

A leukocyturia< 104/mL indicates that there is no inflammatory reaction, which often makes it possible to exclude a urinary tract infection, except in neutropenic subjects or in the case of an early infection.

1.2.6 Red blood cells:

Red blood cells or "red blood cells" are cells without a nucleus that circulate in the blood. They are used to transport oxygen to the cells and reject carbon dioxide. Red blood cells are essential for life. Each red blood cell contains hemoglobin, an iron-rich protein. Hemoglobin is a protein that transports oxygen from the lungs to the body's various organs and removes carbon dioxide from the organs to the lungs. Hemoglobin provides the red color of blood. Hemoglobin contains 65% of the body's iron stores. The calculation of red blood cells allows the detection of certain pathologies such as anemia[7].

Figure 1.7: Red blood cells under electron microscope

Observed under the microscope, a red blood cell reveals its shape: a biconcave disk (disk where the central area is thinner than the periphery) of 7.5 micrometers in diameter and 2 micrometers thick.

The 1st picture is of red blood cells under electron microscope, the 2nd under optical microscope.
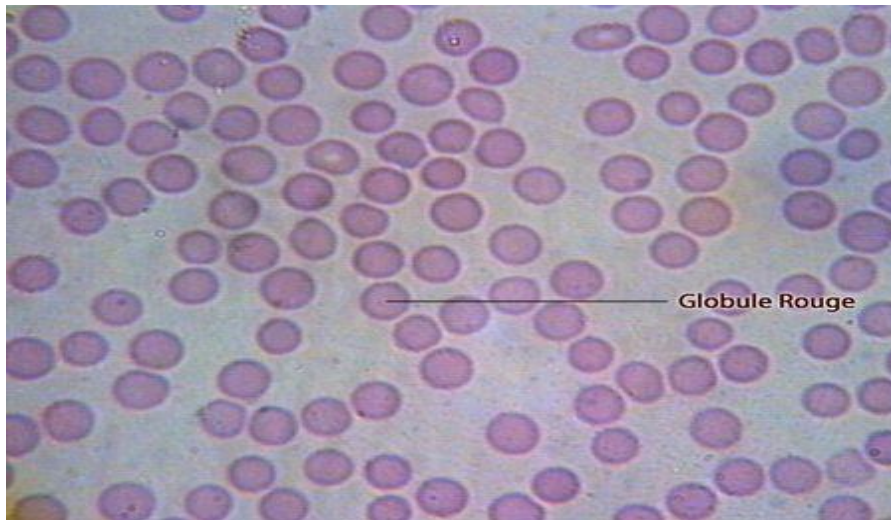
Figure 1.8: Red blood cells under optical microscope.

Hematuria > 104/mL is abnormal. Intact red blood cells have a high probability of coming from the bladder or urethra. Altered red blood cells come from the kidney.

Hematuria is seen in the following conditions:

- ✓ Haemorrhagic forms of nephritis: in this case, haematic cylinders can be found on examination of the sediment;
- ✓ Glomerular damage ;
- ✓ Haemorrhagic cystitis, whether tubercular, gonococcal or with common germs

✓ Renal tuberculosis;

Hematuria may also be associated with other conditions of non-infectious etiology: renal lithiasis, bladder tumors.

1.2.7 Epithelial cells:

The epithelium corresponds to the cellular tissue covering either the external surface of the organism, skin and mucosa, or the internal cavities. It is composed of epithelial cells of various shapes and sizes.
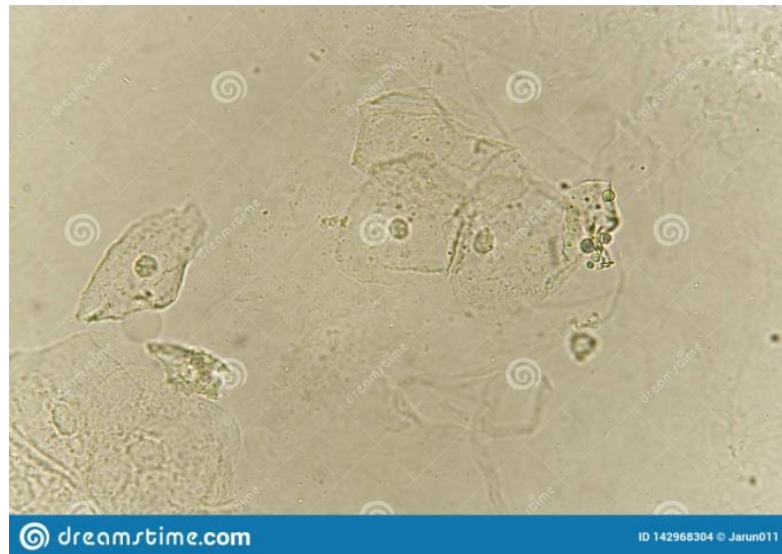
Figure 1.9: Epithelial tissue in a urine sample under a laboratory microscope.

A high number of epithelial cells and the presence of germs attached to them (called "clue cells") are the signature of a bacterial infection. Different bacteria can cause a vaginal infection, whether sexually transmitted or not. The germs involved are identified after culture on suitable media[8].
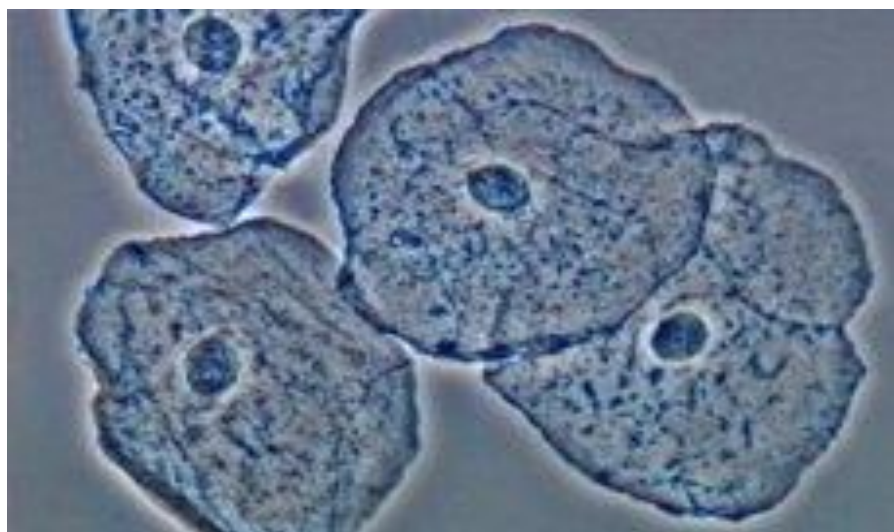
Figure 1.10: Epithelial cells under a laboratory microscope in a urine sample

1.2.7.1 Cylinders:

Cylinders are large elements that are shaped like a part of the renal tubules.

1.2.7.1.1 Formation of Cylinders:

During nephritis or pyelonephritis, the failing glomeruli leak proteins (which are usually absent from the urine).

In the first stage, a protein mold is formed and the proteins are precipitated when the acidity and concentration of the urine reach sufficient values. At this stage, exclusively protein cylinders appear in the urine: the hyaline cylinders[9].

From the protein mold, other types of cylinders can be formed depending on the addition of various cells present in the renal tubule to the basic material:

- ✓ Leukocytes: leukocyte cylinders ;
- ✓ Red blood cells: haematic cylinders;
- ✓ Epithelial cells: epithelial cylinders (much more rare).
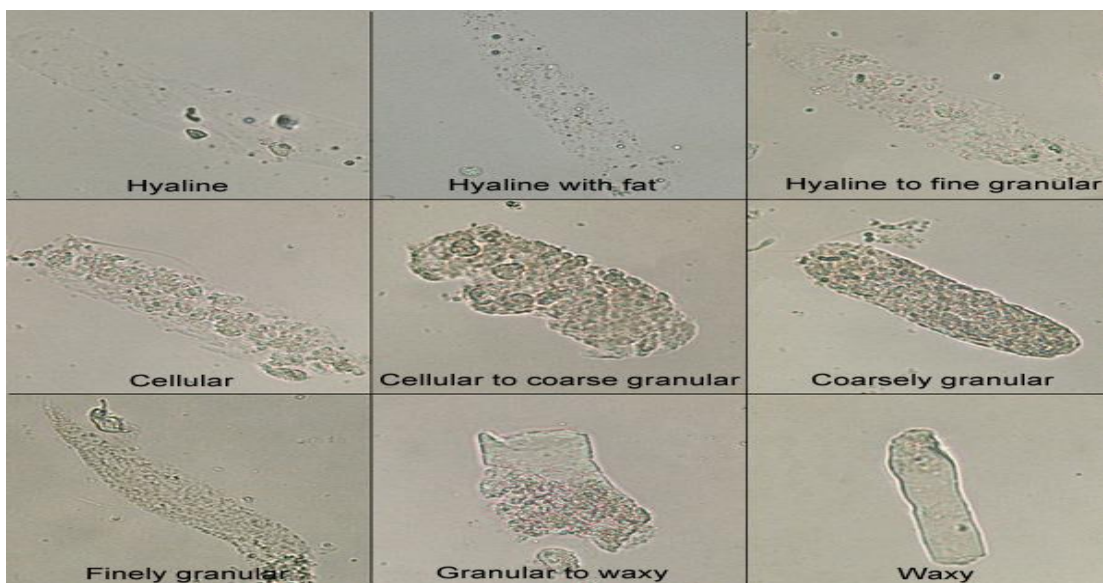
Figure 1.11: Types of cylinders under light microscope in urine sample

1.2.8 Urinary Crystals:

The analysis of urinary crystals by microscopic examination is a simple examination that can provide valuable information.

In the majority of cases, the presence of crystals does not correspond to a pathological condition and, depending on the pH of the urine, urates, uric acid crystals or calcium oxalates can be detected in acidic urine and phosphates, calcium carbonates or ammonium urates in alkaline urine. Phosphates, oxalates and urates are very common[10].

1.2.8.1 Crystals without pathological significance:

1.2.8.1.1 in acidic urine:

Amorphous urates (calcium, magnesium, sodium or potassium urates).They has a sandy appearance, with brownish or orange granules, sometimes grouped in clusters. They are dissolved by alkalization or heat. Normal in moderate quantities, they are increased during gout and in patients undergoing chemotherapy[10].
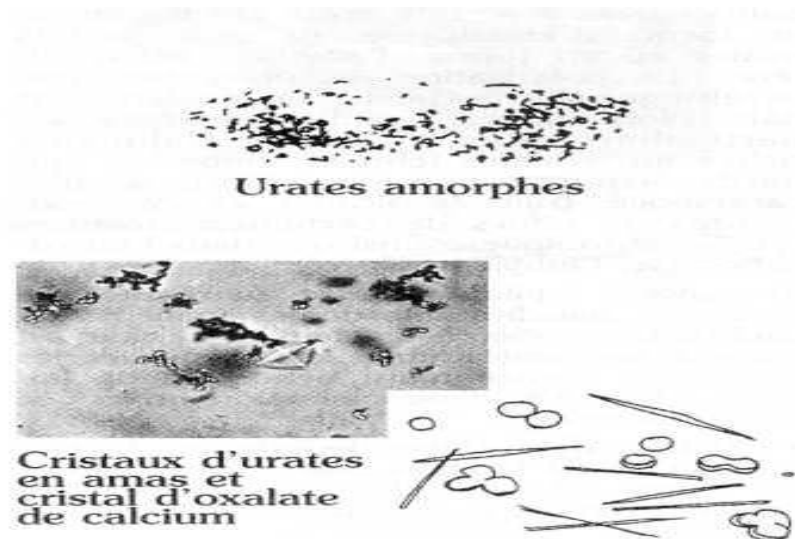
Figure 1.12: Amorphous urates and Calcium oxalate Crystals under microscope optic

Uric acid crystals they are found when the pH is between 5 and 5.5. Colored yellow or reddish-brown, sometimes greenish, their shapes are very varied: quadrangular, flat, oval, lemon-shaped.

Figure 1.13: Oxalate calcium illustrations

Calcium oxalate crystals: They appear in urine of pH 6 to 7, in the form of small colorless octahedrons, small squares with a diagonal cross, in the shape of a shell. They are sometimes oval or peanut or dumbbell shaped. They are increased during certain chronic kidney diseases, as well as in ethylene glycol intoxication[10].

1.2.8.1.2 In alkaline urine :

The main crystals present in alkaline urine are phosphates. Calcium (brushite) and magnesium phosphates are less soluble than sodium and potassium phosphates. All are dissolved when the pH is acidic, especially when dilute hydrochloric acid is added.
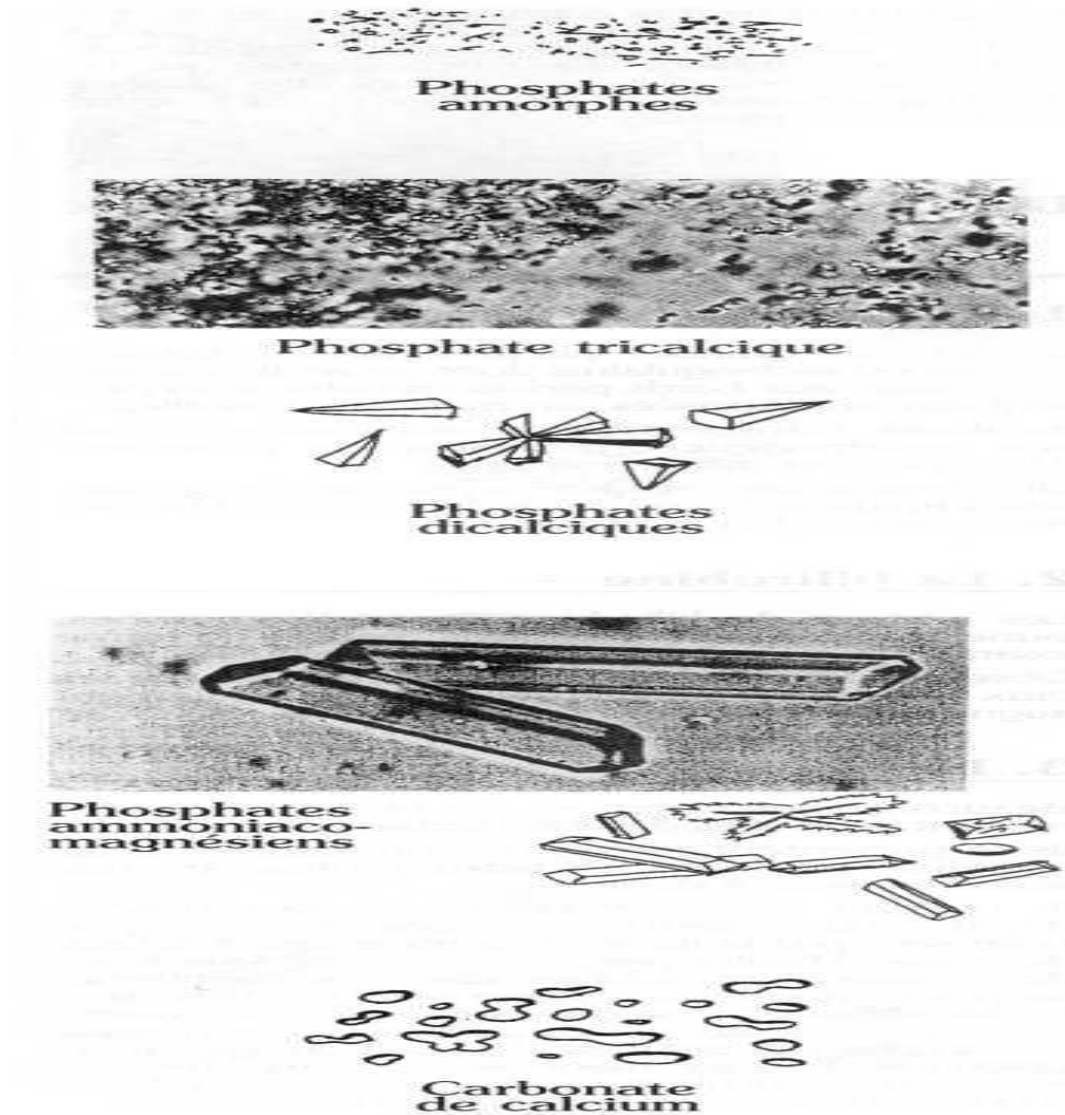
Figure 1.14: some illustrations of urinary crystals and under a microscope

1.3 Crystals corresponding to a pathological state of the patient:

1.3.1 Cholesterol crystals:

These are large thin plates, square or rectangular lamellae, one corner of which is often broken, with sometimes crenellated edges, they can be superimposed, like steps. Colorless, transparent or pale yellow, they have a dull gray or greenish yellow birefringence more or less marked[10].

These crystals, together with lipid droplets, are seen in lympho-urinary fistulas with chyluria (filariasis).

1.3.2 Bilirubin:

Bilirubin crystals are pleiomorphic, yellow-brown, often granular, sometimes diamond-shaped or in needle-like clusters.Observed during liver disease, blood levels of bilirubin are also increased, and urine is acidic[10].

1.3.3 Cystine:

Cystine crystals are found in urine at acidic pH.They appear as colorless, refractive hexagonal plates, sometimes in pairs.They can be confused with uric acid crystals, sometimes hexagonal, but they do not polarize light, unlike uric acid. On the other hand, although both types of crystals are soluble in ammonia, only cystine crystals are soluble in dilute hydrochloric acid (pH < 2). The presence of these crystals is of great clinical value since it is almost pathognomonic of cystinuria. Very rarely it can be observed in cases of heavy metal intoxication or during renal acidosis of tubular origin[10].

1.3.4 Tyrosine:

Tyrosine crystals are characteristic, in the form of more or less fine needles, grouped in clusters, sprays or tufts. They are colorless or pale yellow-greenish, but appear dark or black due to their accumulation. In polarization, they are bright birefringent. They are particularly clear after refrigeration of the urine. They are poorly soluble and are precipitated in alcohol[10].

1.3.5 Leucine :

Leucine crystals appear as droplets, or small yellow spheres, resembling lipids, sometimes bristling with fine points. Radial and concentric striations can be observed, with a characteristic central nest. They can be differentiated from lipids because they are birefringent. They are not soluble in ether. They are soluble in acids and alkalis, they are precipitated in alcohol[10].

1.3.6 Hippuric acid:

These crystals may be present in alkaline urine. They appear as colorless or pale yellow prisms or large needles. They are very refractive, unlike tyrosine crystals.

These crystals are observed during certain intoxications with organic solvents (toluene), and during certain metabolic diseases[10].
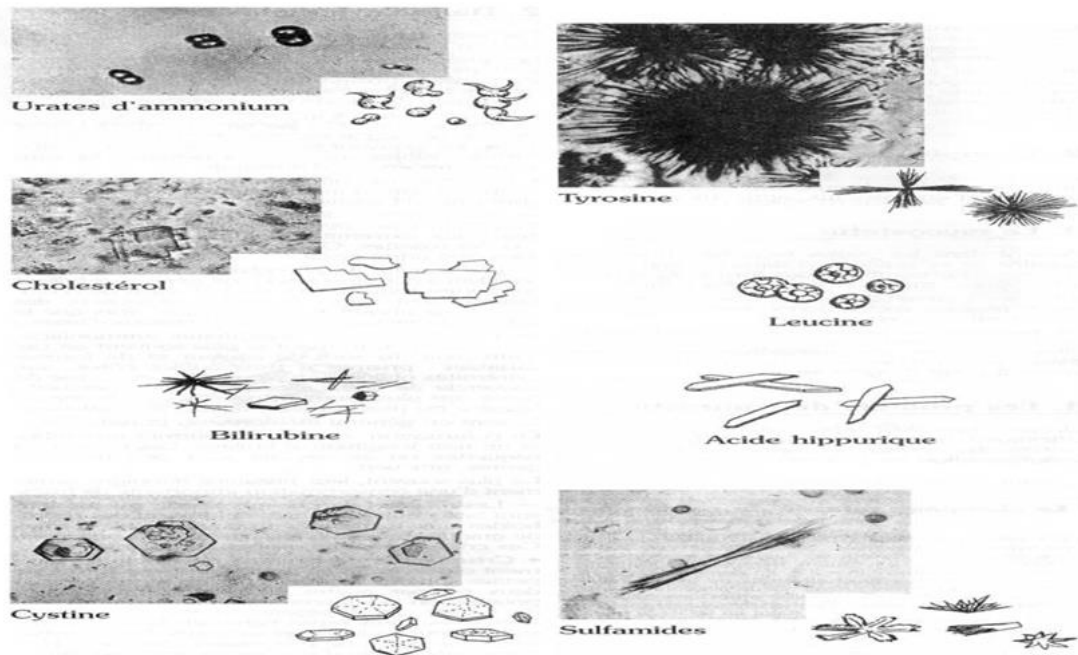
Figure 1.15: Crystal's illustrations corresponding to a pathological state and under microscope

The analysis of urinary crystals can provide valuable information. The identification of these crystals is sometimes easy, however in other less typical cases; one will have to remain very careful about their identification.

1.4 Bacteriological Examination:

1.4.1 Microscopic examinations:

1.4.1.1 Fresh state:

The search for germs under the microscope is done simultaneously with the determination of the leukocyturia in a counting cell. An estimate of the bacteriuria is then possible but the result is very approximate and does not allow differentiating between dead and living bacteria. As yeasts grow slowly on the usual media, their observation on direct examination invites to inoculate adapted media[11].

1.4.1.2 Gram staining:

It lacks sensitivity and for urine with low bacteriuria (≤ 104 CFU/mL), it is necessary to observe the centrifugation pellet. This staining, whether on the noncentrifuged urine or on the centrifugation pellet, is rarely used. However, it has certain advantages:

- ✓ to identify if there are several types of bacteria;

✓ To identify germs that does not grow well on ordinary media (streptococci, corynebacteria). Thus, the observation of Gram + shells in diplococci and short chains or coryneform bacilli must lead to an additional isolation on blood agar.

1.4.2 Bacteriuria or UGC (= Urine Germ Count):

1.4.2.1 Interest for the diagnosis:

Urine germ count (UGC) is a reliable criterion for the diagnosis of infection if all precautions have been taken during urine collection, transport and storage (see paragraph 7.1). In fact, low levels of contamination from bacteria present in the urethral meatus result in bacteriuria that is much lower than that of urinary tract infections[12].

1.4.3 Isolation:

1.4.3.1 Choice of media:

Most urinary tract infections are monomicrobial and due to enterobacteria, i.e., non-demanding bacteria. A NON-SELECTIVE basal medium is usually sufficient.

If these media are inoculated according to the calibrated loop technique, the isolation and enumeration of urinary tract germs will be simultaneous.

1.5 Interpretation of UEC results:

The interpretation of the UEC is difficult and mainly takes into account 5 criteria:

> ➢ Leukocyturia,
> ➢ Bacteriuria,
> ➢ The number of species isolated,
> ➢ Theuropathogenicity class of the isolated species
> ➢ The clinical context.

Urinary tract infections are most often monomicrobial.Bimicrobial community infections are rare in the case of bimicrobial urine, there are two possibilities:

- The presence of a pathologically significant germ and a contaminant, usually from the skin, vaginal or intestinal flora
- The presence of two pathologically significant germs.

Often a second CBEU will allow a decision to be made. A prognosis can however be based on the proportions of each germ:

- If one germ is in the majority, it is likely that the second is a contaminant;
- If the two germs are in equal proportions, bimicrobial infection is most likely.

In all cases, the presence of two types of colonies must be reported and described in the report. And for the clinical context special consideration should be given to:

- Symptoms.
- The presence of a risk factor (urinary catheterization, urinary tract surgery).
- The presence of a possible antibiotic treatment in progress.
- A pregnancy.

1.5.1 Cerebrospinal fluid :

Cerebrospinal fluid, also known as cerebrospinal fluid, is the fluid in which the central nervous system, brain and spinal cord is immersed.It is a clear and colorless biological fluid and its composition is partially constant. It is secreted and reabsorbed continuously and is completely changed 3 to 4 times in 24 hours[13].

The cerebrospinal fluid circulates in two different compartments:

> ➢ The external system: the subarachnoid space which is a space external to the central nervous system located in the spine, and which surrounds the spinal cord;
> ➢ The internal system: intraencephalic compartment made of ependymal cavities or ventricle.

In case of bacterial or viral meningitis:

- The diagnosis can only be affirmed by the examination of the CSF which must be done in emergency.

- Meningitis is always a topical disease. The sudden nature of this disease means that the mortality rate does not decrease. However, almost all the germs responsible for purulent meningitis are very sensitive to the "new antibiotics".

- To reduce this mortality rate, the diagnosis and therapeutic management of meningitis must be accelerated.

- Therefore, the examination of a CSF in the laboratory is an urgent examination.

1.5.1.1 CSF collection and transport:

CSF is collected, ideally before any antibiotic therapy, by lumbar puncture, after rigorous asepsis of the skin, between the lumbar vertebrae L3-L4 or L4-L5 (Figure 1.17)

It is collected in a minimum of 3 sterile tubes numbered 1, 2 and 3.

The total volume should be between 2 and 5 mL. An additional 1 to 2 mL should be used for additional tests such as mycobacteria.

- Tube 1 for biochemical analysis.
- Tube 2 for cytological analysis.
- Tube 3 for culture.
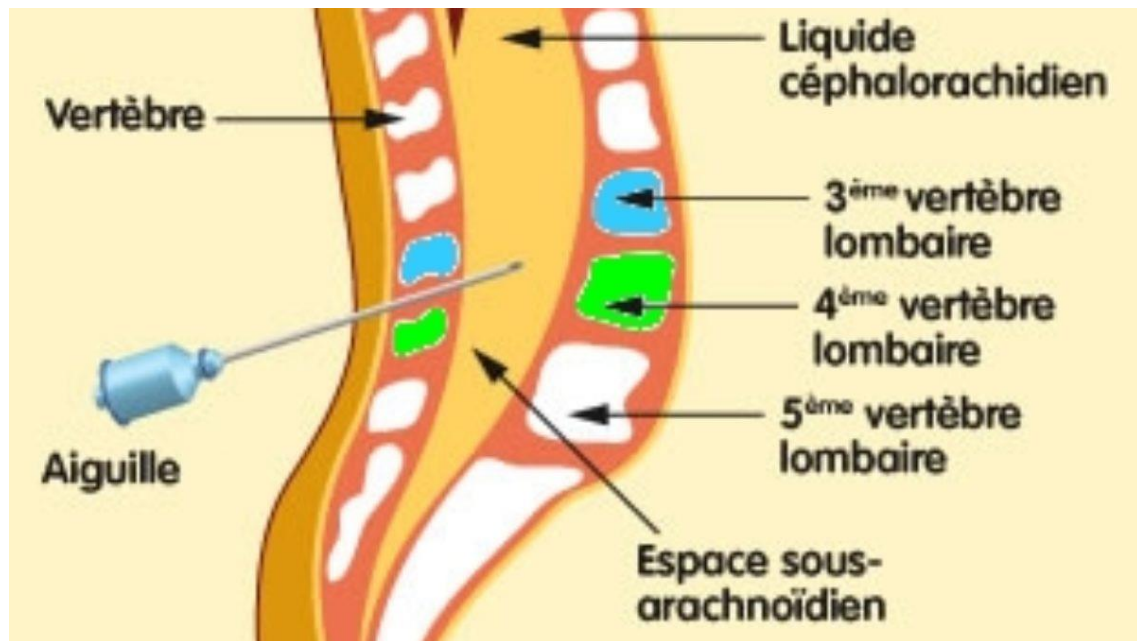
Figure 1.16: Illustration of the lumbar puncture

A minimum of information must be provided to the laboratory:

- Current antibiotic treatment.
- Age.
- Immune status and type of immunosuppressant.
- The clinical and epidemiological context: convulsion, purpura, cases in the family, neurosurgery or ENT surgery...

As a general rule, a blood culture is performed in parallel.
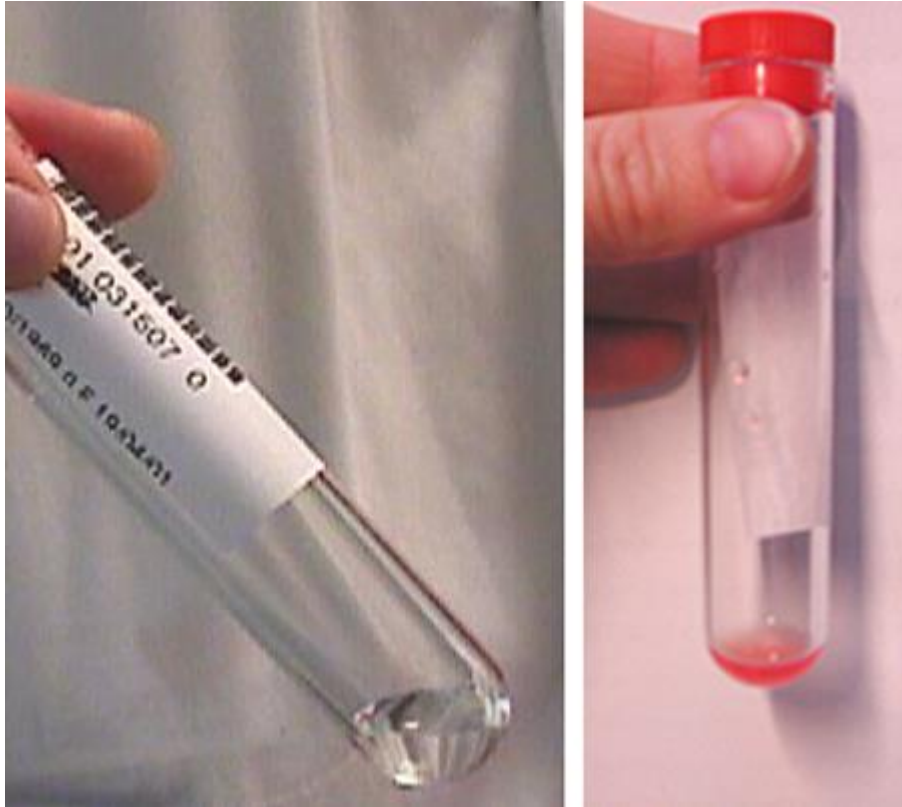
Figure 1.17: Cerebrospinal fluid (left: normal, clear, "rock water." Right: hemorrhagic during subarachnoid hemorrhage).

1.5.2 Steps in the CSF Examination:

1.5.2.1 Gross examination:

Because the diagnosis of meningitis is urgent, this gross examination is often crucial. Indeed, the observation of cloudy CSF is sufficient to diagnose meningitis.

CSF may present in the following ways:

1.5.2.1.1 Cloudy CSF:

A turbid fluid is always pathological and indicates, in most cases, meningitis of bacterial etiology and acute evolution. Its observation by the person performing the puncture is therefore very important because it implies the immediate initiation of a probabilistic antibiotic treatment without waiting for the results of the antibiogram. In the case of community meningitis, an anti-inflammatory treatment with corticosteroids accompanies it in order to reduce the risks of mortality and the frequency of auditory and neuromotor sequelae[13].

1.5.2.1.1 Clear CSF:

- Either it is normal
- Or it is meningitis with clear liquid; the etiological agents are very varied.
- Listeria monocytogenes ;

- Tuberculous mycobacteria (most often Mycobacterium tuberculosis = bacillus Koch or BK).
- Viruses.
- Cryptococcusneoformans .
- Treponema pallidum (syphiliticmeningitis).
- Leptospirainterrogans.
- Lyme disease: occurs a few weeks or months after an infesting tick bite (inoculating Borreliaburgdorferi).

CSF from bacterial meningitis decapitated by antibiotic treatment can also be clear.

Finally, CSF is clear in non-infectious meningitis such as neoplastic meningitis and drug-induced meningitis.

Citrine yellow CSF = xanthochromic

The yellow color of the cerebrospinal fluid is explained by the transformation of hemoglobin into hematogenous pigments during a meningeal hemorrhage.

1.5.2.1.3Hemorrhagic CSF:

Two causes are possible:

➢ A meningeal hemorrhage accompanying or not a bacterial meningitis .
➢ Or a rupture of a vessel during sampling.

To distinguish these two possibilities, one can:

- Collect CSF in 3 different tubes. Only the first tube (or even the second) is bloody in the case of a vessel rupture during sampling

- Or centrifuge the CSF: the supernatant is xanthochromic in the case of meningeal hemorrhage.

Finally, cytology and biochemical analysis of the CSF allow to distinguish a meningeal hemorrhage complicating a bacterial meningitis from a meningeal hemorrhage without meningitis

1.5.3 Placing in culture:

Media inoculated systematically, whatever the appearance of the CSF and its leukocyte concentration .Media allowing the growth of fastidious bacteria responsible for purulent meningitis is inoculated.

1.5.4 Cytological examination:

1.5.4.1 Leukocyte count (on non-centrifuged CSF):

It is not necessary for turbid fluids because the turbidity indicates a number of leukocytes greater than 200 per mm3.After gentle agitation of the CSF, the count is performed in a counting cell. Normal CSF contains less than 5 leukocytes per mm3 (10 to 30 mm3 in newborns).In clear fluid meningitis, the leukocyte count is greater than 100/mm3.

1.5.4.2 Leukocyte count:

This is established after cytocentrifugation and May-Grunwald-Giemsa staining.In the absence of a cytocentrifuge, a smear is prepared from a drop of the centrifugation pellet and spread like a blood smear.

A Position de chargement (Arrêté)
B En cours d'exécution (centrifugation)
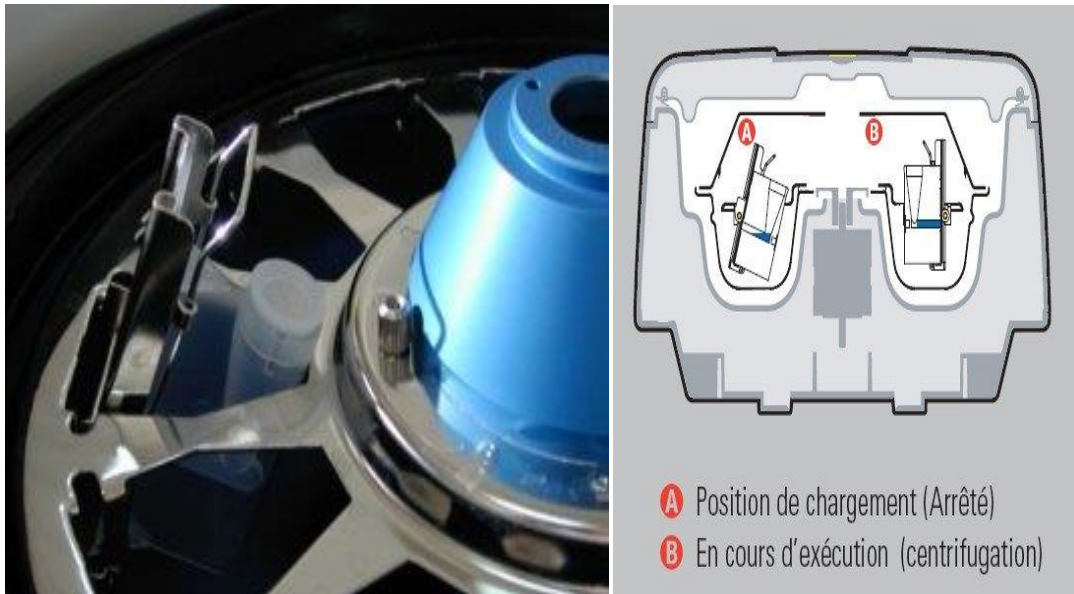
Figure 1.18: The cytocentrifuge

In the case of cloudy fluids, a strong majority of granulocytes is observed (80 to 100%) except in Listeria meningitis for which the 3 types of cytologies are observed (predominance of granulocytes, predominance of lymphocytes, mixed formula).

In the case of clear fluid meningitis, a predominance of lymphocytes points to tuberculous meningitis and viral meningitis.

Note: a formula with more than 10% eosinophilic granulocytes in the CSF is observed in cases of helminthiasis (Angiostrongyluscantonensis, Gnathostomaspinigerum,...), leukemia or following drug treatments[14].

1.5.5 Biochemical parameters useful for diagnosis:

Biochemical tests of CSF and serum are useful in guiding the diagnosis.

1.5.5.1 Biochemical examination of CSF :

- Glycorrhaphy (glucose assay) .
- Protein sputum (protein assay).
- Lactate assay.

- Interferon alpha assay.

1.5.5.2 Biochemical examination of serum :

- Blood glucose.
- Procalcitonin assay.
- CRP assay.

1.5.5.3 Glycorrhaphy/blood glucoses ratio :

Glycorrhaphy should always be expressed as a function of blood glucose. A normal glycorrhaphy is equal to 65% of the blood glucose.Hypoglycorachy is indicative of bacterial meningitis.Hypoglycorrhaphy is explained by the reduced transfer of glucose across the blood-meningeal barrier and the consumption of glucose by the inflammatory reaction.

1.5.5.4 Proteinorachy:

Normal values for the protein pool are between 0.10 and 0.45 g/L (although the protein pool in neonates and infants under 2 months of age is higher).

A very high protein count is indicative of bacterial meningitis. It varies from 1 to 5 g/L in purulent meningitis.Proteinorachy may also be slightly increased in viral meningitis, without exceeding 1.5 g/L.Hyperproteinorachia may persist for 2 to 3 weeks after the start of treatment[15].

1.5.5.5 Determination of lactate in CSF:

Lactate is a sign of anaerobic metabolism. For a while it was abandoned, but a recent study has given it renewed interest.

Indeed, it is observed that during bacterial meningitis the concentration of lactate in the CSF is always higher than 3.2 mmol/L.

1.5.5.6 Determination of interferon alpha in CSF:

Interferon alpha, a cytokine of innate immunity produced only in the presence of viruses, is used as an early marker of viral meningoencephalitis.

1.5.5.7 Determination of procalcitonin and CRP in serum:

Procalcitonin is an inflammatory protein secreted by the thyroid gland and whose serum concentration increases very rapidly following bacterial aggression. This marker allows to quickly distinguishing viral meningitis from bacterial meningitis

➢ Liquid effusion example pleural fluid

A pleural fluid effusion is the presence of fluid in the pleural cavity, i.e. between the two layers of the pleura, the visceral layer adhering to the lung and the parietal layer adhering to the rib cage. Usually, this space is virtual.

Depending on the size of the effusion, it can be demonstrated clinically by lung auscultation and percussion, and radiologically on a chest x-ray.

A pleural puncture allows the patient to be relieved and, after analysis of the fluid, to know the origin of the effusion[16].
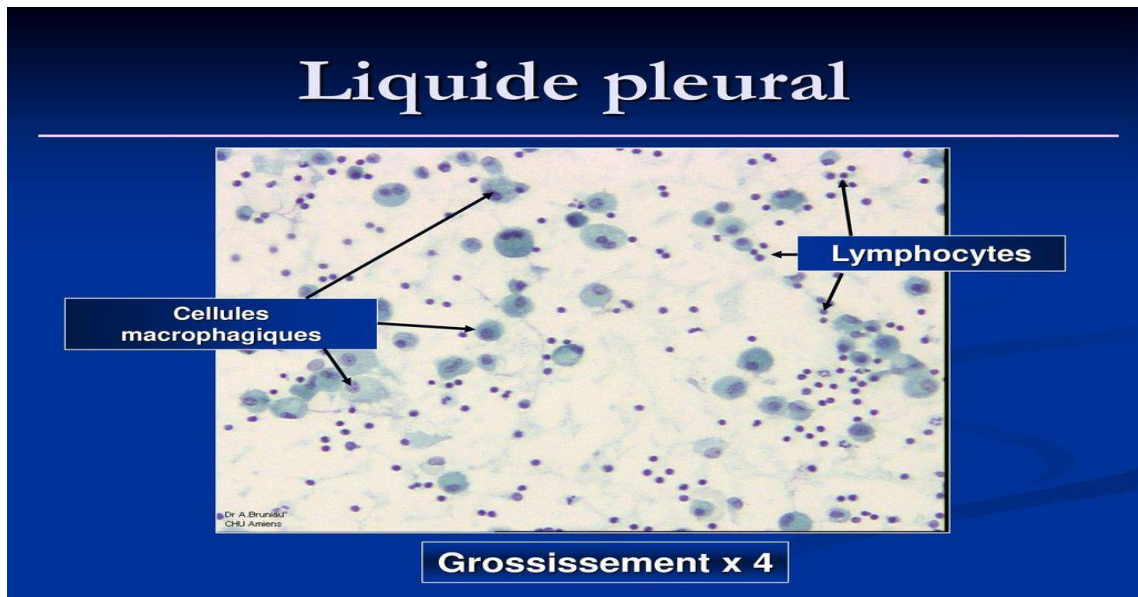
Figure 1.19: Macroscopic appearance of pleural fluid

	Transudates	Exudates
Pleural proteins / Serum proteins> 0.5	–	+
Pleural LDH / Serum LDH> 0.6	–	+
Pleural LDH> 2/3 upper limitof normal serum LDH level	–	+

Table 1.1: Distinctive criteria for transudates and exudates

There are six categories of effusions:

- Transparent lemon appearance characteristic of transudate
- Dark yellow translucent appearance is in favor of an exudate. The more protein it contains, the darker yellow the fluid is, and the more cells it contains, the more opalescent it is.
- Cloudy yellow or more characteristically greenish yellow appearance synonymous with purulent pleurisy.
- Oserohemorrhagic, characterized by a pink or red appearance of the fluid in relation to the presence of red blood cells. This fluid does not clot in the syringe and has a hematocrit of only a few percent.
- Frankly hemorrhagic appearance (hemothorax) showing the accumulation of blood in the pleural cavity, most often in connection with a vascular wound.
- More exceptional but easily recognizable, milky effusion in case of chylous effusion (chylothorax)

- Note:

When the protidemia is within normal limits, a transudate has a protid concentration of less than 30 g/L, whereas exudate usually contains more than 30 g/L of protid.

- Pleural protide/serum protide ratio > 0.5; pleural fluid LDH concentration > 200 IU/l; and/or pleural LDH/serum LDH ratio > 0.6. An exudate meets at least one of the conditions, whereas a transudate meets none. They are a valuable aid in the orientation or confirmation of the etiology of the effusion.

Etiologic of transudates	Etiologic of exudates
✓ Heart failure ✓ Pulmonary embolism ✓ Hepatic cirrhosis ✓ Nephrotic syndrome ✓ Constrictive pericarditis ✓ Urinothorax ✓ Petritoneal dialysis ✓ Myxedema	✓ Infections ✓ Pulmonary embolism ✓ Hemothorax ✓ Neoplasia ✓ Digestive diseases ✓ Latrogene ✓ Chylothorax ✓ End-stage renal disease

Table 1.2: Etiologies of transudates and exudates

1.6 Cytological examination :

1.6.1 Red blood cells:

- when the red blood cell count is less than 100 RBCs/mL (100,000/mm3), no diagnostic value
- when this level is higher than 100 RBC/ml, it may be useful to ask for a pleural hematocrit determination to distinguish hemothorax (Ht> 20%) from other causes of hemorrhagic pleurisy (mainly cancer, but also pulmonary embolism, trauma, asbestosis and Dressler syndrome)

1.6.2 White blood cells :

- A transudate generally contains less than 1000 cells/mm3 and
- The highest cellularities (> 50,000 cells/mm3) are found in purulent pleurisy
- A formula is used to distinguish between pleurisy with a predominance of neutrophils, lymphocytic pleurisy and pleurisy with eosinophils (> 10%)

- The presence of atypical cells on cytological examination may point to a diagnosis of neoplastic pleurisy, but pleural histology is always necessary to make a definite diagnosis of malignancy.

Appearance of the liquid	Cell analysis	Etiology
Clear liquid $\Longrightarrow$	GB<1000/mm^3 Proteins<30g/l $\Longrightarrow$	heart failure +++,nephritic syndrome, drrhosis
Bleeding fluid (hemothorax) $\Longrightarrow$	GR>10000/mm^3 Ht>20% OR >50%Ht of peripheral sg $\Longrightarrow$	traumatic context or not
Serohaemaqtic fluid $\Longrightarrow$	GR>100000/mm^3 20%>Ht>1% $\Longrightarrow$	EP++ Cancer++ Mild trauma
Purulent fluid Altered PN $\Longrightarrow$	GB>50000/mm^3 $\Longrightarrow$	Purulent pleurisy
Straw yellow liquid $\Longrightarrow$	GB between 50 $\nearrow$ 80% of PN $\Longrightarrow$	parapneumatic pleurisy+++ very early stage tuberculosis, pancreatitis,subphrenic abscess, Ep
	Et 10000/mm^3 $\longrightarrow$ 50% of A $\Longrightarrow$	tuberculous pleurisy +++ cancer, lymphoma and CLL connectivitis
>10% eosino	$\searrow$ $\Longrightarrow$	EP, anterior pleural puncture, drugs, parasitosis, vasculitis, idiopathic pleurisy.
Milky liquid Chytomicrons+++ $\Longrightarrow$	triglycerides>1g/l $\Longrightarrow$	

Table 1.3: Etiological orientation according to the macroscopic appearance and the result of the cytological analysis of the pleural fluid

Predominance of neutrophiles $\Longrightarrow$	**Purulent pleurisy**
	Parapneumonic pleurisy
	Pancreatic
	Subpherical abscess
	Pulmonary embolism
	Tuberculosis
Rheumatoid pleurisy>50% $\Longrightarrow$	**Tuberculosis**
	Cancer
	Lymphoma
	Pleural lymphocytosis
Pleural eosinophilia>10% $\Longrightarrow$	**Pulmonary embolism**
	Hemothorax (> 8 days)
	Drug-induced pleurisy
	Parasitosis and histoplasmosis
	Asbestosis
	Vasculitis
	Repeated pleural punctures
	Idiopathic pleurisy

Table 1.4: Etiologies of exudates depending on the leukocyte formula of pleural fluid

1.7 Conclusion:

As we've seen in this chapter different microscopic tests but the same pathological ways is to detect and approximate or count the pathogens in different mediums under the microscope, in the next chapter we will cover different methods of detection and recognition involving Deep learning algorithms that will help speed up the process and reduce error.

Chapter 2 Neural Networks architectures and models

2.1 Introduction :

For us to detect and classify the pathogens we sited in the above chapter we need to use Artificial Intelligence exactly Deep Learning.

In this chapter we will define and present basic understanding of ANN, Perceptron, MLP, CNN also some models such as VGG and Yolo.

2.2 Artificial Neuron ANN:

The artificial neural network is one of the most sophisticated artificial intelligence algorithms in the world. This algorithm was originally inspired by biological functioning. This algorithm is capable of learning any task: driving a car, playing chess, or classifying and recognizing images. In other words, artificial neural networks are one of the main tools of machine learning and are systems inspired by the human brain that aim to reproduce the ways in which humans learn. The latter consists of input and output layers and, in most cases, contains a hidden layer consisting of units that transform the input into something that can be useful for the output layer. These tools are used to find patterns that are too complex for a human programmer to extract and teach the machine to recognize[17].

2.3.1 the perceptron:

The perceptron is a supervised learning algorithm for binary classifiers (i.e. separating two classes). It was invented in 1957 by Frank Rosenblatt at the Cornell University Aeronautics Laboratory. It can be seen as the simplest type of neural network. It is a linear classifier. This type of neural network does not contain any cycles (feedforward neural network). In its simplified version, the perceptron is single-layer and has only one output to which all inputs are connected. It is a formal neuron with a learning rule that automatically determines the synaptic weights in order to separate a supervised learning problem. If the problem is linearly separable, a theorem ensures that the perceptron rule can find a separator between the two classes. In the figure below, we can observe the perceptron. It has n inputs (X1 to Xn), these inputs are generally integers or real numbers transmitted to the perceptron. To each of the X inputs is

associated a weight W which will be used to compute the output Y. Finally, a bias connection is added, as it is necessary for the proper functioning of the perceptron[18]. The calculation performed to obtain the output Y is the following:

$$Y = \sum W_n X_n \geqslant \theta$$

When it comes to binary classification, we will say that if Y is greater than θ, the class is worth 1 otherwise it is worth zero.
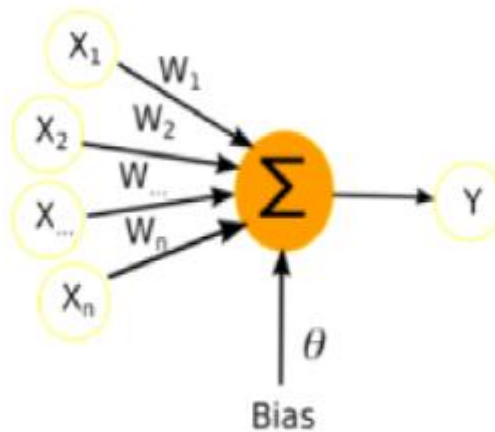

Figure 2.1: simple perceptron diagram

2.3.1.1 Perceptron training

Training a perceptron is relatively simple. Suppose we have a base B of k examples, each example is composed of n measures X1 to Xn which are real numbers and an output c which is the result we want the perceptron to obtain. The algorithm that trains the perceptron is the following:

Input: A base consisting of B of k examples

1- Random initialization of all the weights of the network

2- Random selection of an example i from the base B

3-Calculation of the output Y for this example with the formula

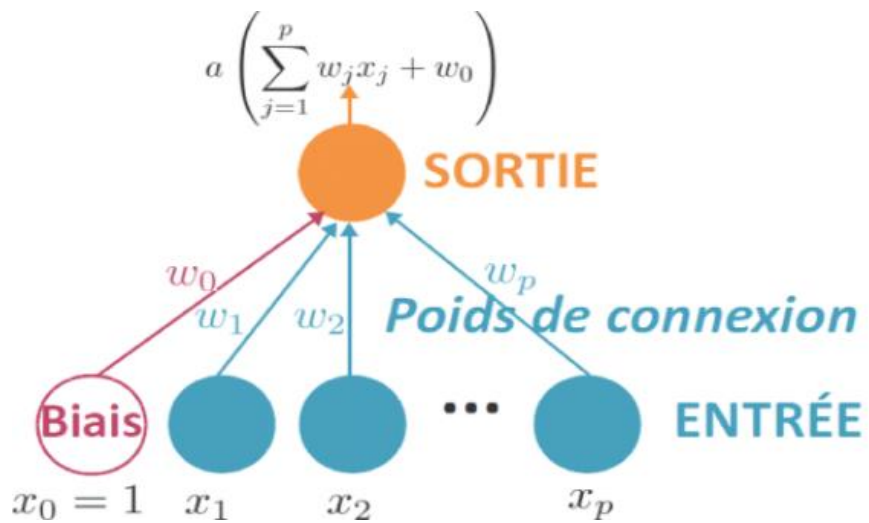4- Updating the weights using the formula: $W\_j = W\_j + (c - Y) * x\_j$

Figure 2.2: perceptron characteristics diagram

2.3.2 The multilayer perceptron (MLP):

The multilayer perceptron is the first neural network to have found many practical applications such as flower recognition, fraud detection, etc.. It can be used for all supervised classification tasks. Nowadays, it is one of the most popular models, and is implemented by many libraries such as TensorFlow, Weka, Scikit-Learn, etc[19].
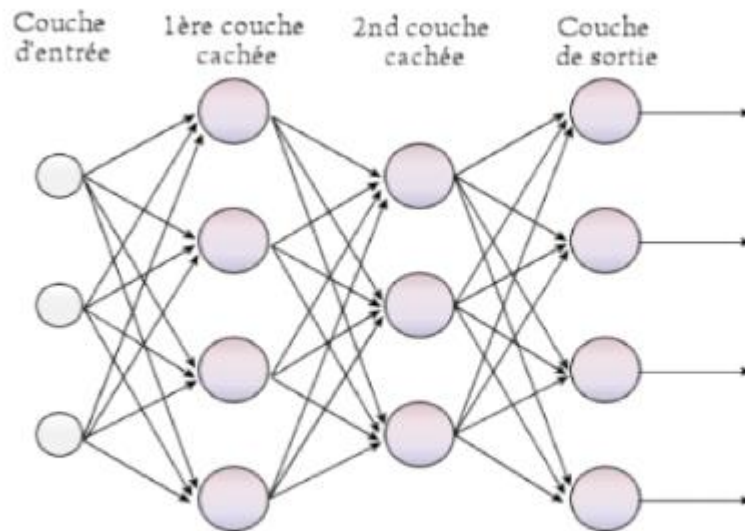

Figure 2.3: multilayer perceptron scheme

We can see the neurons of the multilayer perceptron as a multitude of perceptron connected to each other. the topological feature of this network is that all the neurons of a layer are connected to all the neurons of the following layer. Each neuron has n inputs, n being the number of neurons in the previous layer, and an output which is sent to all the neurons of the next layer. The calculation of the output of a neuron is done according to the activation function that has been chosen.

2.4 Learning:

Learning is a phase in the development of a neural network during which the behavior of the network is modified until the desired behavior is obtained.
learning :
- Supervised learning
- Unsupervised learning

2.4.1 Supervised learning:

2.4.1.1 definition of supervised learning :

Supervised learning, in the context of artificial intelligence, is the most widely used learning method in Machine Learning and Deep Learning. Supervised learning consists in supervising the machine's learning by presenting it with examples of what it should do. It has many uses: voice recognition, artificial intelligence
le, classifications, etc. Thus, linear regression is one of the most widely used supervised learning techniques for predicting a continuous value. Also, the vast majority of machine learning and deep learning problems use supervised learning: it is therefore essential to correctly understand how this method works.

2.4.1.2 Supervised learning works in 4 steps:

- Import a Dataset (x,y) that contains our examples

- Develop a Model with random parameters

- Develop a Cost Function that measures the errors between the model and the

  Dataset

- Develop a Learning Algorithm to find the model parameters that minimize the

  Cost Function

a. The Dataset:

Data Science starts with the word... Data!
The first step of a Supervised Learning algorithm is therefore to import a Dataset containing the examples that the machine must study.

This dataset always includes 2 types of variables:

- **A** *target* variable y
- One or more *feature* variables x

b. The Model:

The model is a simplified representation of reality, that we can use to predict what would happen under certain conditions. It can be a drawing, a physical equation, a mathematical function, a curve... in short, any representation. A Machine Learning model is not based on a mathematical proof or a physical equation. Instead, it is built from data, like a statistical model. If for example your dataset gives you the following scatterplot, then the machine will have to find the model that best fits in this scatterplot[20].
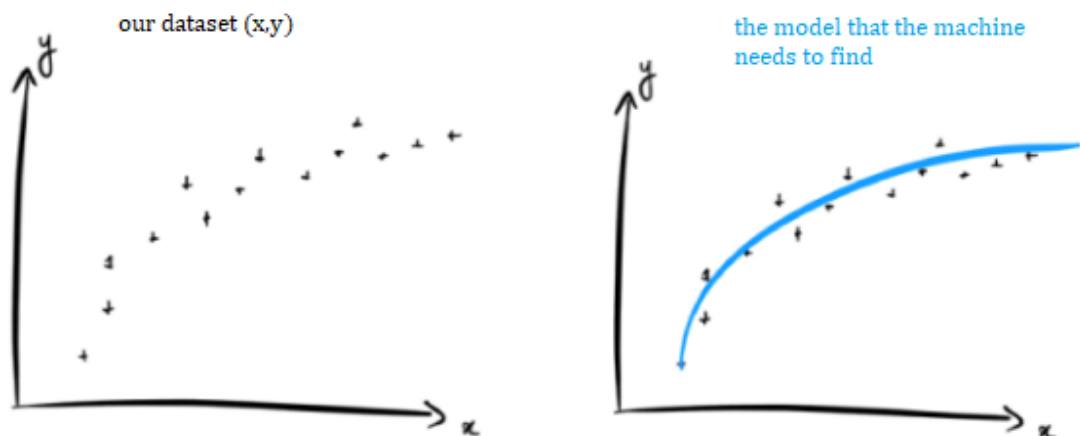

Figure 2.4: plot sketch of model after training on a dataset

c.   :

For the machine to find the best model, it must already be able to measure the performance of a given model in Machine Learning. To know which model is the best among 2 candidates, we have to evaluate them. For that, we measure the error between a model and the dataset, and we call it the Cost Function[21].

2.4.2 Unsupervised machine learning:

Unsupervised Learning consists in having only input data (X) and no corresponding output variables.
The goal of unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.
It is called unsupervised learning because, unlike supervised learning above, there is no correct answer or teacher. The algorithms are left to their own devices to discover and present the interesting structure of the data.

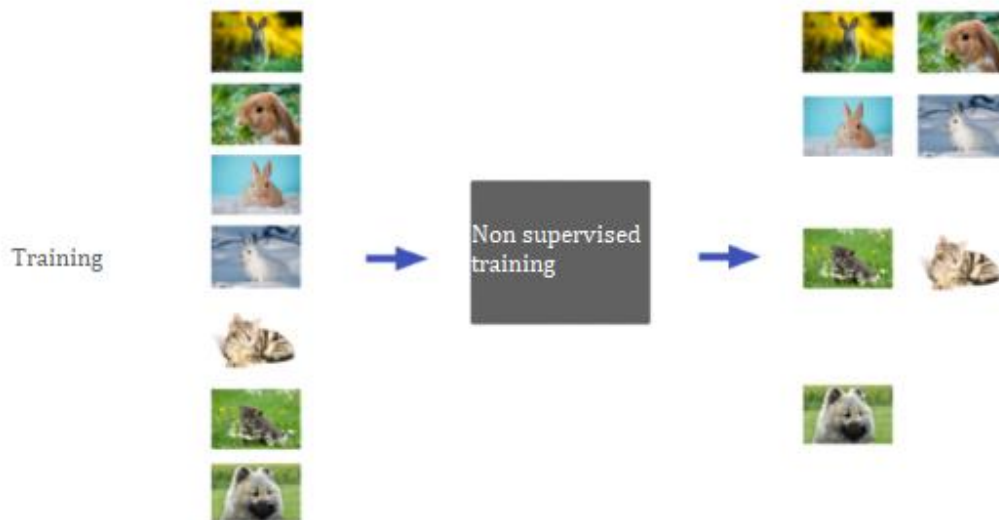Unsupervised learning includes two categories of algorithms: Clustering and association algorithms[22].

Figure 2.5: Example of Unsupervised Machine Learning

2.4.2.1 Clustering:

Clustering is the process of separating or dividing a dataset into a number of groups, so that datasets belonging to the same groups are more similar than those belonging to other groups. In simple terms, the goal is to separate groups with similar traits and assign them into clusters[23].

2.4.2.2 Association:

Association is about discovering interesting relationships between variables in large databases. It discovers the probability of co-occurrence of items in a collection.

In summary, clustering is about grouping data points based on their similarities, while association is about discovering relationships between the attributes of these data points.

Here is a list of some unsupervised machine learning algorithms:

- K-means clustering

- Dimensionality Reduction

- Neural networks / Deep Learning

- Principal Component Analysis

- Singular Value Decomposition

- Independent Component Analysis

- Distribution models

- Hierarchical clustering

2.4.3 Semi-supervised automatic learning:

Problems where you have a large amount of input data (X) and only some of the data is labeled (Y) are called semi-supervised learning problems. Therefore, these problems fall between supervised and unsupervised learning.

2.4.4 Reinforcement learning:

Reinforcement learning is learning by interacting with the environment and observing the results of certain actions. It allows machines to automatically determine the ideal behavior in a specific context, in order to maximize its performance. For this, a simple feedback of the results is necessary to learn how the machines should act. This is called the reinforcement signal.

The term "cause and effect" for reinforcement learning can be characterized by the following steps:

1. The agent observes an input state
2. An action is determined by a decision making function (policy)
3. The action is performed
4. The agent receives a result according to its environment
5. Information about the result given for this state or action is recorded

By performing actions, we observe the resulting rewards to determine the best action for a given state.

2.4.4.1 Agent exploitation and exploration:

A learning agent is subject to the trade-off between exploitation (repeating actions, which it knows will give it good rewards) and exploration (trying new actions, to learn new things).
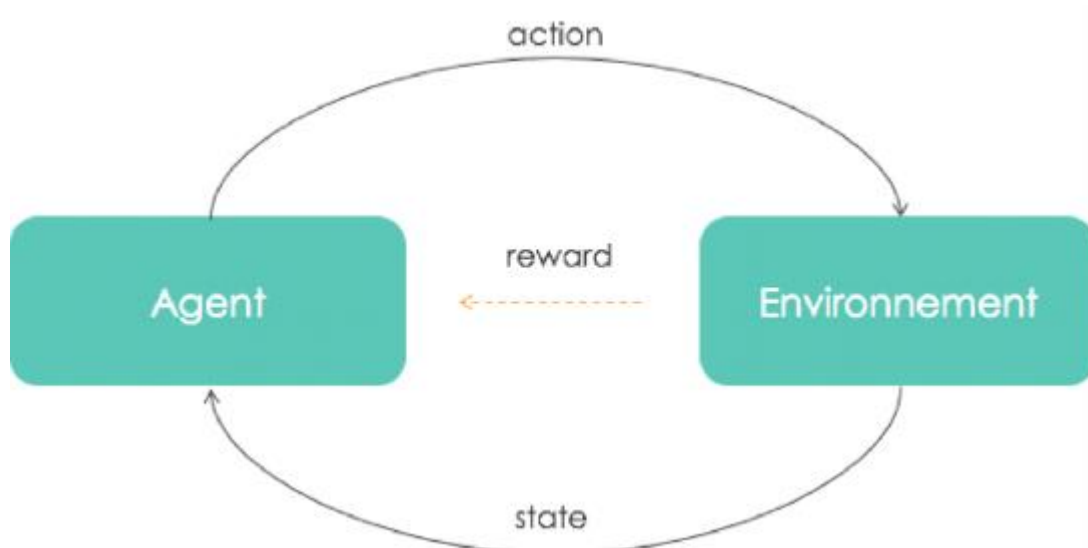

Figure 2.6: diagram of agent exploitation and exploration

2.4.4.2 Operation :

To understand how reinforcement learning works, we can compare it with training a pet. There are indeed similarities in the process.
Pets do not understand human language, so it is impossible to teach them tricks by verbally telling them what to do. Instead, it is possible to provoke a reaction from the animal by creating a situation.
If the animal's reaction is the right one, it can be rewarded. Thanks to the rewards, the pet will be gradually conditioned. As soon as he is confronted with the same situation, he will be able to perform the same Action.
Thus, we can compare the pet to an AI agent. The environment it is exposed to is your home, and the words spoken act as a trigger to move it from one state to another.

2.4.4.3. Algorithms:

a. Model-based and model-free:

The algorithm is model-based if it takes the model of the environment as input. Typically, the algorithm takes the Markovian decision process as input. In particular the algorithm has access to the transition function and the probabilities. The algorithm has access to $p(s', r \mid s, a)$, the probability of being in state s' and of having the reward from state s by executing action a. On the contrary, an algorithm is model-free if it does not use a model as input. The algorithm does not use probabilities $p(s', r \mid s, a)$ because it does not know them. On the other hand of course, a model-free algorithm has data structures for states and actions.

b. on-policy VS off-policy

The algorithm is on-policy when it evaluates and improves the policy, which is the same one used to make decisions during learning. The algorithm is off-policy if the evaluated and improved policy is different from the one the agent uses to make decisions during learning. We then distinguish the target policy, which is the learned policy, from the behavior policy. Off-policy algorithms are generally slower to converge. On the other hand, off-policy algorithms are more generalizable (on-policy algorithms are finally off-policy where the target policy and the behavior policy are the same). Off-policy algorithms can be used when episodes are generated by a non-conventional controller, or by a human expert.

2.5 Activation functions:

A neuron is simply a function with parameters.

This function is $(output = activation(w\_1. x1 + w\_2. x2 + ... + w\_n x\_n))$.

The input is provided by the data, or by the outputs of the previous neurons. It is the vector $((x\_1, x\_2, .. x\_n))$

The weights of the neuron are the values $((w\_1, w\_2, .. w\_n))$. These are the only values that are specific to the neural network. When we save our trained network on the disk, these weights will be recorded. The weights of a neural network, together with its architecture (number of layers, number of neurons on each layer), define it entirely.

The output is a real number, given by the above function.

And the activation function is a non-linear function. We will see below the different activation functions usually used.

If we use the identity function ($(f(x) = x)$), then the neuron will represent a simple linear combination of inputs and weights.

In general, non-linear functions are used. Two of the most common functions are the sigmoid, and reLu :
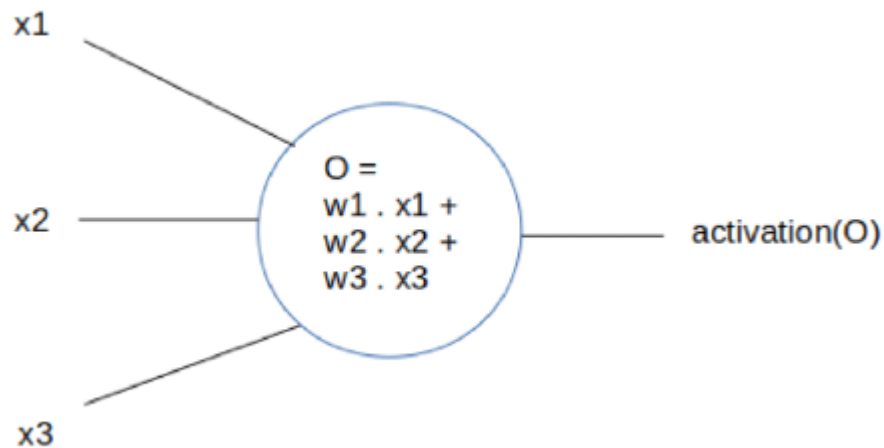

Figure 2.7: diagram of perceptron with the activation function in output
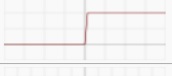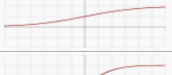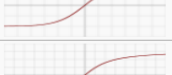
2.5.1 The different types of activation functions:
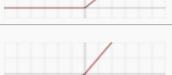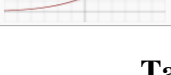
Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for} & x < 0 \\ 1 & \text{for} & x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for} & x \neq 0 \\ ? & \text{for} & x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \dfrac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \dfrac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \dfrac{1}{x^2 + 1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for} & x < 0 \\ x & \text{for} & x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for} & x < 0 \\ 1 & \text{for} & x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) [2]		$f(x) = \begin{cases} \alpha x & \text{for} & x < 0 \\ x & \text{for} & x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for} & x < 0 \\ 1 & \text{for} & x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) [3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for} & x < 0 \\ x & \text{for} & x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for} & x < 0 \\ 1 & \text{for} & x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \dfrac{1}{1 + e^{-x}}$

Table 2.1: different Activation functions

2.6 Deep learning:

Deep learning is one of the main technologies of machine learning. With Deep Learning, we are talking about algorithms capable of mimicking the actions of the human brain through artificial neural networks. The networks are composed of dozens or even hundreds of "layers" of neurons, each receiving and interpreting information from the previous layer.

2.6.1 CONVOLUTIONAL NEURAL NETWORK (CNN) :

Convolutional networks are a particular form of multilayer neural network whose architecture is inspired by the visual cortex of mammals. These artificial neural networks are capable of categorizing information from simple to complex.

2.6.1.1 Architecture of a CNN:

A convolution neural network can have several convolution steps, ReLu and Pooling. A rule to respect is that the ReLu function must be applied after a convolution step in order to have a non-linear response, but Pooling is not mandatory.

After going through all the convolution, ReLu and pooling steps, we can move on to image classification. The last phase consists in sending all the pixels to a multilayer neural network. Since we have been able to recover the most important parts of an image that we have condensed, the classification phase will be much more efficient than using an artificial neural network without convolution[24].

38

A CNN is simply a stacking of several layers of convolution, pooling, ReLU correction and fully-connected. Each image received as input will be filtered, reduced and corrected several times, to finally form a vector. In the classification problem, this vector contains the probabilities of belonging to the classes.

All convolutional neural networks must start with a convolution layer and end with a fully-connected layer. The intermediate layers can be stacked in different ways, provided that the output of one layer has the same structure as the input of the next. For example, a fully-connected layer, which always returns a vector, cannot be placed before a pooling layer, since the latter must receive a 3D matrix.

In general, a neural network stacks several convolution and ReLU correction layers, then adds a pooling layer (optional), and repeats this pattern several times; then, it stacks fully-connected layers[24].

2.6.1.3 The different types of layers of a CNN:

a- Convolution layer (CONV): Convolution, from a simplistic point of view, is the application of a mathematical filter to an image. From a more technical point of view, it is about dragging a matrix over an image, and for each pixel, using the sum of the multiplication of that pixel by the value of the matrix. This technique allows us to find parts of the image that could be interesting[25].
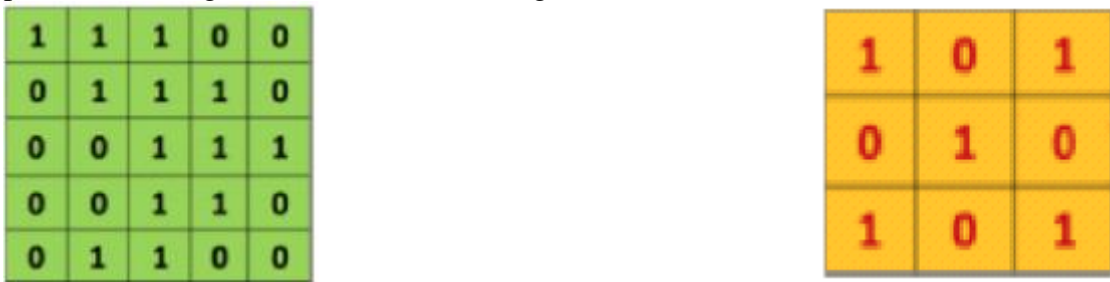

Figure 2.8: 5x5 image and 3x3 filter kernel

The values should vary between 0 and 255 and are represented by 1's and 0's. In a real case, these values are continuous and can be positive or negative.

Apply the filter on the image: in the image matrix M, we can see that each pixel value of the tile image (the orange boxes) is multiplied by each corresponding filter value (1x1, 1x0, 1x1 ....). Then add all these values to obtain a single value '4' which will be part of a new convolved image.
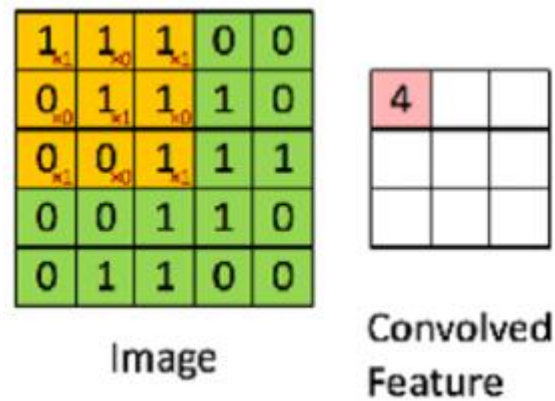
Figure 2.9: convolution pixel wise example

The filter must move one square with each iteration until the first row is finished. When we have finished the first row, the filter "moves down" one square and the same procedure is repeated for each row and column.
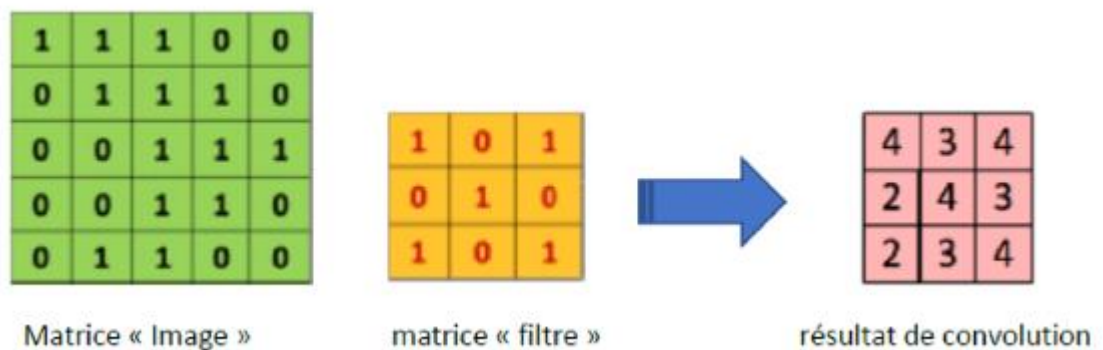

Matrice « Image »   matrice « filtre »   résultat de convolution

Figure 2.10: example of applying 3x3 conv filter on 5x5 image

Noted that a 3x3 convolution of depth 1 performed on a 5x5 input feature map, also of depth 1. Since there are nine possible 3x3 slots to extract tiles from the 5x5 feature map, this convolution generates a 3x3 output feature map.
A convolution neural network contains multiple filters and these filters are applied on the original image. After the first step we have as many new images as filters. The convolution phase can also be seen as hidden layers of neurons where each neuron is connected to only a few neurons in the next layer.

To explain how a filter can find interesting parts in images. Let's take the previous Filter. The values that are 1 form an "X". During the convolution phase, when this filter is applied to a shape that exactly matches an "X", then the value obtained will be higher. The less the shape of the image matches the shape of the filter, the lower the value

obtained. We can therefore reduce the size of the image and highlight the interesting elements.

In order to demonstrate the effects of a convolution phase, the following (Figure 2.11), which is a synthetic image of building B, will be modified with several mathematical filters. For example, if we want to find edges, we can use the values of the F2 filter.

Figure 2.11: applying a filter to an image

The filters are also adapted to each learning iteration because the values of the mathematical filters used are weights as in multilayer neural networks

b- Correction layer (ReLu- Rectified linear unit):ReLu is a function that must be applied to each pixel of an image after convolution, and replaces each negative value with a 0. If this function is not applied, the created function will be linear and the XOR problem persists since in the convolution layer no activation function is applied.

ReLu is widely used in convolution neural networks because it is a fast function to compute: $(y) = max(0, y)$. Therefore, its performance is better than other functions where costly operations must be performed[26].

c- Layer pooling (POOL): This type of layer is often placed between two convolution layers: it receives as input several feature maps, and applies to each of them the pooling operation. The pooling operation consists in reducing the size of the images, while preserving their important features. To do this, we cut the image into regular cells, and then we keep the maximum value within each cell.

The method used is to imagine a window of 2 or 3 pixels that slides over an image, as for convolution. But, this time, we make steps of 2 for a window of size 2, and steps of 3 for 3 pixels. The size of the window is called "kernel size" and the steps are called "strides". For each step, we take the highest value among those present in the window and this value constitutes a new pixel in a new image. This is called Max Pooling[27].
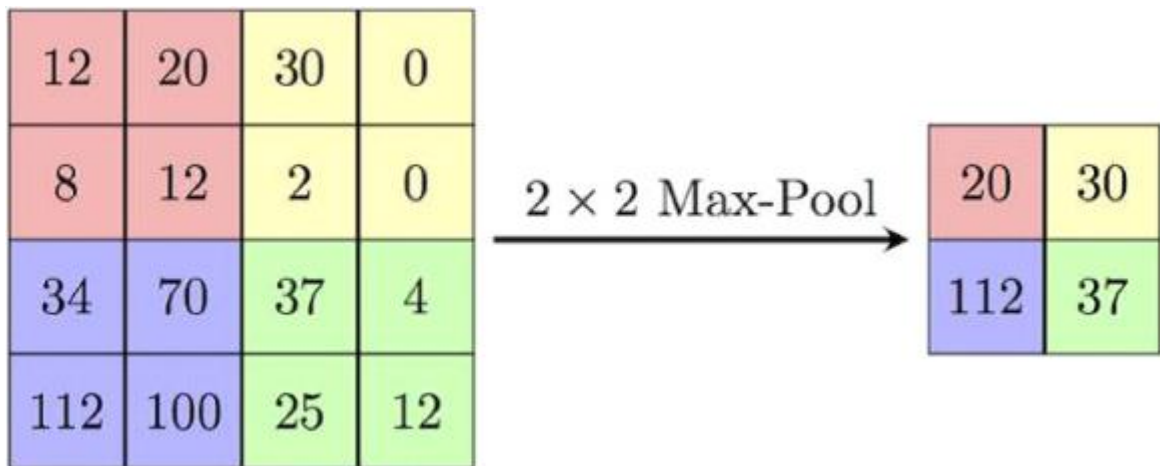
Figure 2.12; Max pool with a 2x2 filter and a step of 2

The pooling layer reduces the number of parameters and calculations in the network. This improves the efficiency of the network and avoids overlearning. The maximum values are located less accurately in the feature maps obtained after pooling than in those received as input, this is in fact a great advantage, Indeed, when we want to recognize a dog for example, its ears do not need to be located as precisely as possible: knowing that they are located roughly next to the head is enough ,Thus, the pooling layer makes the network less sensitive to the position of the features ,the fact that a feature is located a little bit higher or lower, or even that it has a slightly different orientation should not cause a radical change in the image classification.

e. Fully-connected layer: The fully-connected layer is always the last layer of a neural network. This type of layer receives a vector as input and produces a new vector as output. To do this, it applies a linear combination and possibly an activation function to the values received as input.

The last fully-connected layer is used to classify the input image of the network: it returns a vector of size N, where N is the number of classes in our image classification problem. Each element of the vector indicates the probability for the input image to belong to a class.

For example, if the problem is to distinguish cats from dogs, the final vector will be of size 2: the first element (respectively, the second) gives the probability of belonging to the class "cat" (respectively "dog"). Thus, the vector [0.9 0.1] means that the image has a 90% chance of representing a cat.

To compute the probabilities, the fully-connected layer multiplies each input element by a weight, sums it up, and then applies an activation function (logistic if N=2, softmax if N>2): This treatment amounts to multiplying the input vector by the matrix containing the weights. The fact that each input value is connected with all output values explains the term fully-connected[28].

f. Loss layer : The loss layer is the last layer of the network. It calculates the error between the network prediction and the actual value. In a classification task, the random variable is discrete, as it can take only the values 0 or 1, representing the membership (1) or not (0) to a class. This is why the most common and suitable loss function is the cross-entropy function.

It comes from the field of information theory, and measures the global difference between two probability distributions (the one of the model prediction, the one of the real) for a random variable or a set of events . Formally, it is written :

$$Loss(x, class) = - \sum_{class=1}^{C} y_{x,class} Log_{(P_{x,class})}$$

Where y is the estimated probability of x belonging to class i, p is the real probability of x belonging to class i, given that there are C classes.

The information of the receptive field that they process.

2.7 models and algorithms:

2.7.1 VGG:

is a model of convolutional neural network, there are two algorithms available: VGG16 and VGG19, both architectures are very similar and follow the same logic, VGG19 has a greater number of convolution layers.

2.7.1.1-VGG 16:

The model only requires a specific preprocessing which consists in subtracting the average RGB value, computed on the training set, of each pixel.

During model training, the input for the first convolution layer is an RGB image of size 224 x 224. For all convolution layers, the convolution kernel is of size 3×3: the smallest dimension to capture the notions of top, bottom, left/right and center. This was a specificity of the model at the time of its publication. Until VGG16 many models were moving towards higher dimension convolution kernels, these convolution layers are accompanied by Max-Pooling layers, each of size 2×2, to reduce the size of the filters during learning. At the output of the convolution and pooling layers, we have 3 layers of Fully-Connected neurons. The first two are composed of 4096 neurons and the last one of 1000 neurons with a softmax activation function to determine the image class[29].
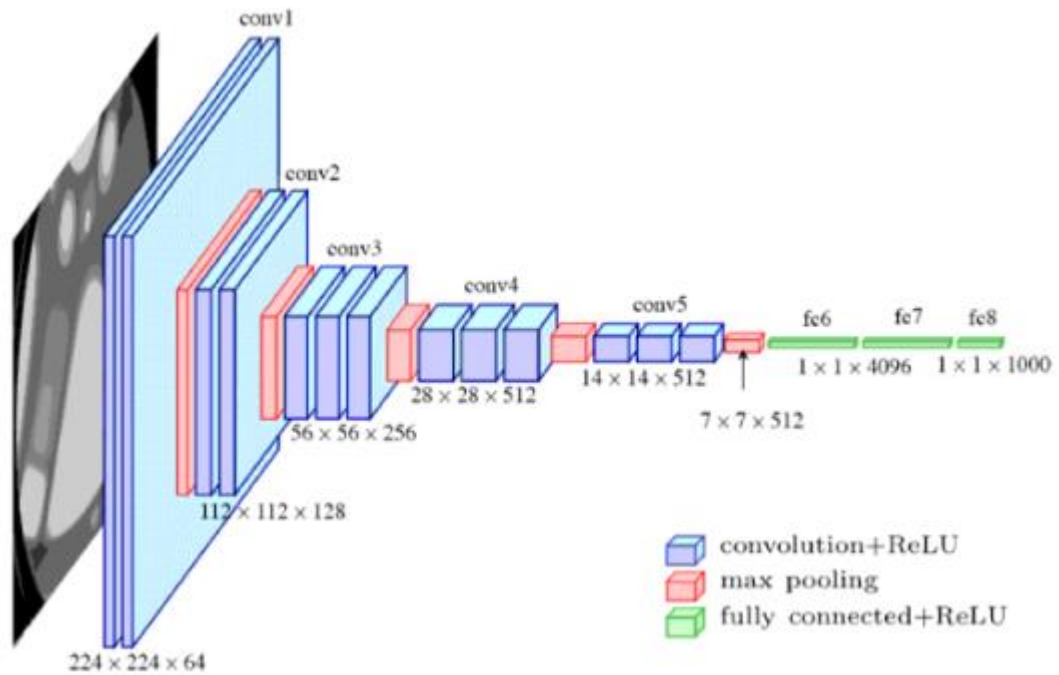
Figure 2.13: visualization of the VGG-16 architecture

2.7.2 YOLO:

The "You Only Look Once," or YOLO, family of models are a series of end-to-end deep learning models designed for fast object detection, developed by Joseph Redmon, et al. and first described in the 2015 paper titled "You Only Look Once: Unified, Real-Time Object Detection."

For YOLO , detection is a simple regression problem that takes an input image and learns the class probabilities and selection frame coordinates. YOLO divides each image into an S x S grid and each grid predicts N bounding boxes and confidence. The confidence reflects the accuracy of the selection frame and whether it actually contains an object (regardless of class) . YOLO also predicts the ranking score of each box for each class in training. You can combine the two classes to calculate the probability that each class is present in a predicted box. Thus, the total number of SxSxN boxes is predicted. However, most of these boxes have a low confidence score and if we set a confidence threshold of 30%, we can remove most of them[30], as shown in the example below:
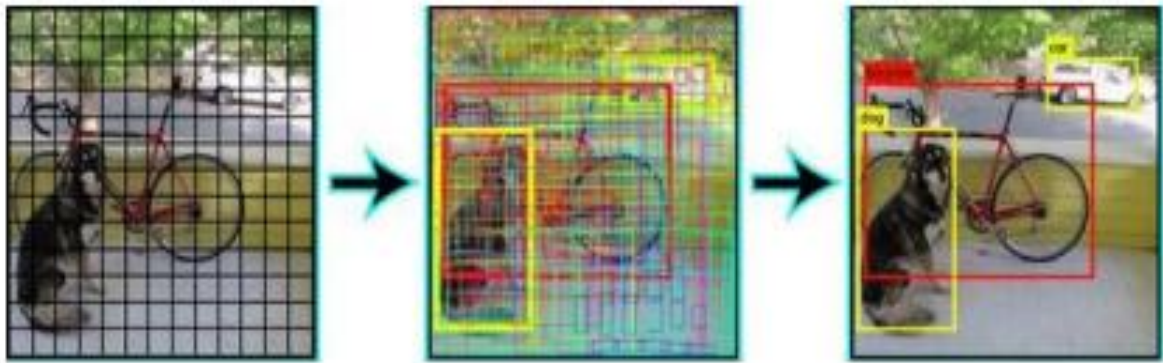
Figure 2.14: Object detection using YOLO algorithm

2.8 Conclusion:

In this chapter we discussed neural networks from a basic perceptron to a deep layer network such us YOLO with explanation of some details.

In the next chapter we will introduce the cloud processing and show how we generate pseudo microscopic images in MATLAB and then train them on the YOLO V5 model and discuss the results.

Chapter 3 Database creation and results

3.1 introduction:

In this last chapter we will implements what we discussed on the chapters above on an cloud environment and we will explain how the pseudo microscopic image generator was realized and also the process of the training using the Yolo v5 model adding up the evaluation of the results and testing on new real microscopic image.

3.2 Software implementation:

This section describes the software implementation, the process of reading data .

3.2.1 Computer tools used:

3.2.1.1 Google Colab:

Google Colab is an executable document. Colab is like a Jupyter Notebook which is already provides some basic libraries. You can write, run and share the Jupyter code within Google Drive and all the Colab projects are stored in Google Drive.

Figure 3.1: Google Colab Logo

A notebook document is composed of cells, each of which can contain code, text, images, and more. Colab connects your notebook to a cloud-based runtime, meaning you can execute Python code without any required setup on your own machine. Additional code cells are executed using that same runtime; you can change the python version and change GPU and TPU at a run time. Colab provides the more useful functionality at a runtime and other compatible frameworks.
[31]

To use Google Colab, just go to Google drive then click on new then on "more" and choose "Colaboratory". If you do not have the "Colaboratory" option, click on "Associate more applications" then search for "Colaboratory" and click on connect. [32]

We can perform the following using Google Colab:

- Write and execute code in Python
- Create/Upload/Share notebooks
- Import/Save notebooks from/to Google Drive
- Import/Publish notebooks from GitHub
- Import external datasets
- Integrate PyTorch, TensorFlow, Keras, OpenCV
- Free Cloud service with free GPU

[33]

3.2.1.2 MATLAB:

MATLAB is commercial interactive calculation software. It allows performing numerical simulations based on numerical analysis algorithms. It can therefore be used for the approximate resolution differential equations, partial derivative equations or linear systems, etc.

Matlab sessions are twofold: knowledge of this software is in itself essential because it is increasingly most used in industry and banks to develop software prototypes and test new algorithms. Every year between twenty and thirty percent of DESS students use Matlab for their internship.

Figure 3.2: MATLAB Logo

Then his learning will go through the practice of the numerical analysis algorithms studied more theoretically in the rest of the module. For that we will realize during the sessions a project directly inspired by a concrete case schematizing an industrial problem. It should be noted in passing that shareware / freeware software is increasingly emulating Matlab.

Satisfactory. Their use overcomes the main drawback of the cost of the license. You can

For example, download Scilab for free from the INRIA2 website you can learn to

Program in Scilab whose syntax is close to that of Matlab with the recent and complete work of G. Allaire and S. M. Kaber3 [34]

3.2.1.3 YOLOv5:

YOLO (You Only Look Once) models are used for Object detection with high performance. YOLO divides an image into a grid system, and each grid detects objects within itself. They can be used for real-time object detection based on the data streams. They require very few computational resources.

History of YOLO

- Yolov1 (Jun 8th, 2015): You Only Look Once: Unified, Real-Time Object Detection

- Yolov2 (Dec 25th, 2016): YOLO9000:Better, Faster, Stronger

- Yolov3 (Apr 8th, 2018): YOLOv3: An Incremental Improvement

- Yolov4 (Apr 23rd, 2020): YOLOv4: Optimal Speed and Accuracy of Object Detection

- Yolov5 (May 18th, 2020): Github repo (there is no paper as of Aug 1st, 2021) [35]

YOLOv5 is a family of object detection architectures and models pretrained on the COCO dataset, and represents Ultralytics open-source research into future vision AI methods, incorporating lessons learned and best practices evolved over thousands of hours of research and development. [36]

As YOLO v5 is a single-stage object detector, it has three important parts like any other single-stage object detector.

1. Model Backbone
2. Model Neck
3. Model Head

Model Backbone is mainly used to extract important features from the given input image. In YOLO v5 the CSP—Cross Stage Partial Networks are used as a backbone to extract rich in informative features from an input image.

CSPNet has shown significant improvement in processing time with deeper networks.

Model Neck is mainly used to generate feature pyramids. Feature pyramids help models to generalized well on object scaling. It helps to identify the same object with different sizes and scales. Feature pyramids are very useful and help models to perform well on unseen data. There are other models that use different types of feature pyramid techniques like FPN, BiFPN, PANet, etc. In YOLO v5 PANet is used for as neck to get feature pyramids. For more information on features pyramids .The model Head is mainly used to perform the final detection part. It applied anchor boxes on features and generates final output vectors with class probabilities, objectness scores, and bounding boxes. In YOLO v5 model head is the same as the previous YOLO V3 and V4 versions.
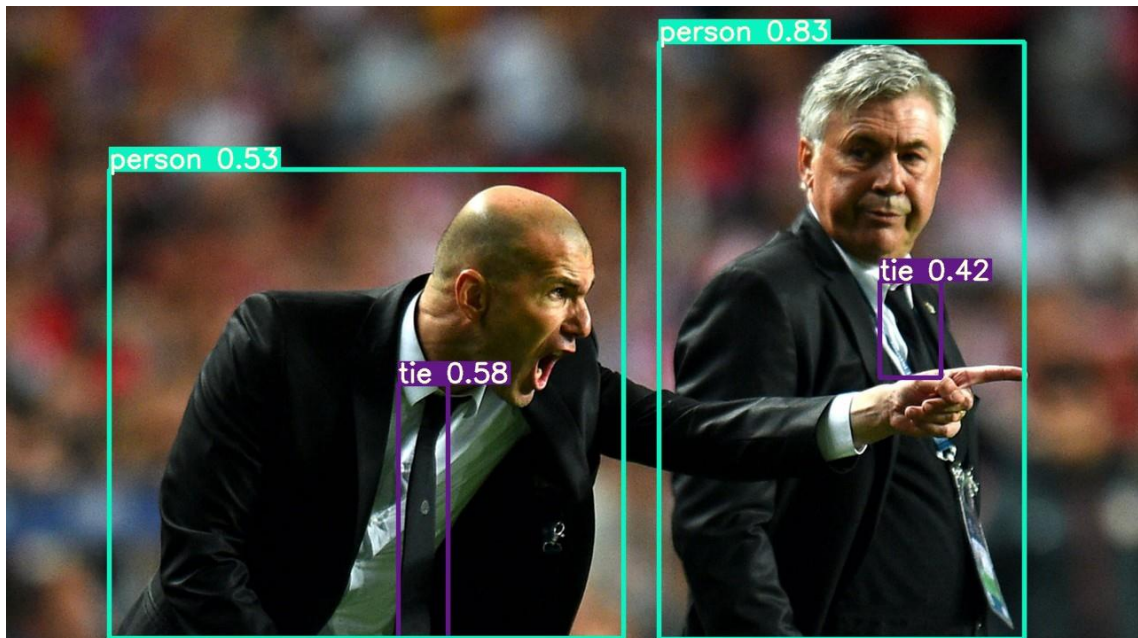
Figure 3.3: YOLOv4 VS YOLOv5 [37]

3.3 Pseudo Microscopic Image Generator:

In this section we will discuss all about the Pseudo microscopic images generating process.

3.3.1 Pathogens images:

Due to the lack of datasets for all the different pathogens we needed to make our own datasets but with only few images that we did collect across the internet.

Figure 3.4:different types of pathogens with transparent background

All the images were 69 by 69 and we made the following numbers:

- Bacterial 25 images.
- Hippuric Acid 21 images.
- Oxalate Calcium 35 images.
- Yeast 32 images.
- Epithilial 28 images.
- Red Blood Cells 45 images.
- White Blood Cells 42 images.

With four different background that are extracted from microscope images that contains no particles.
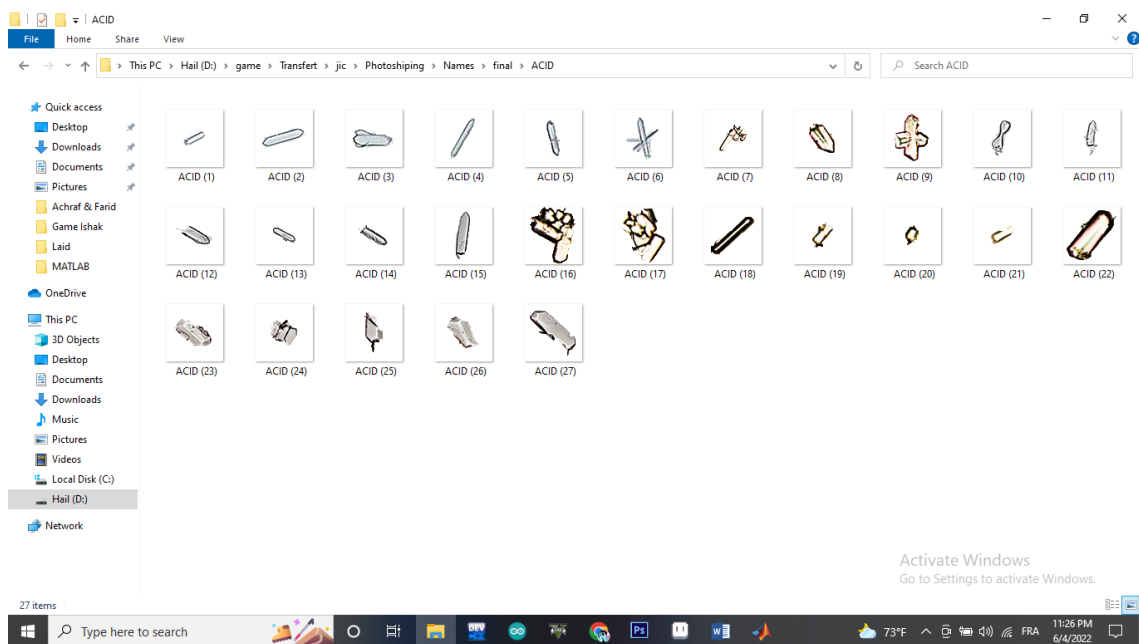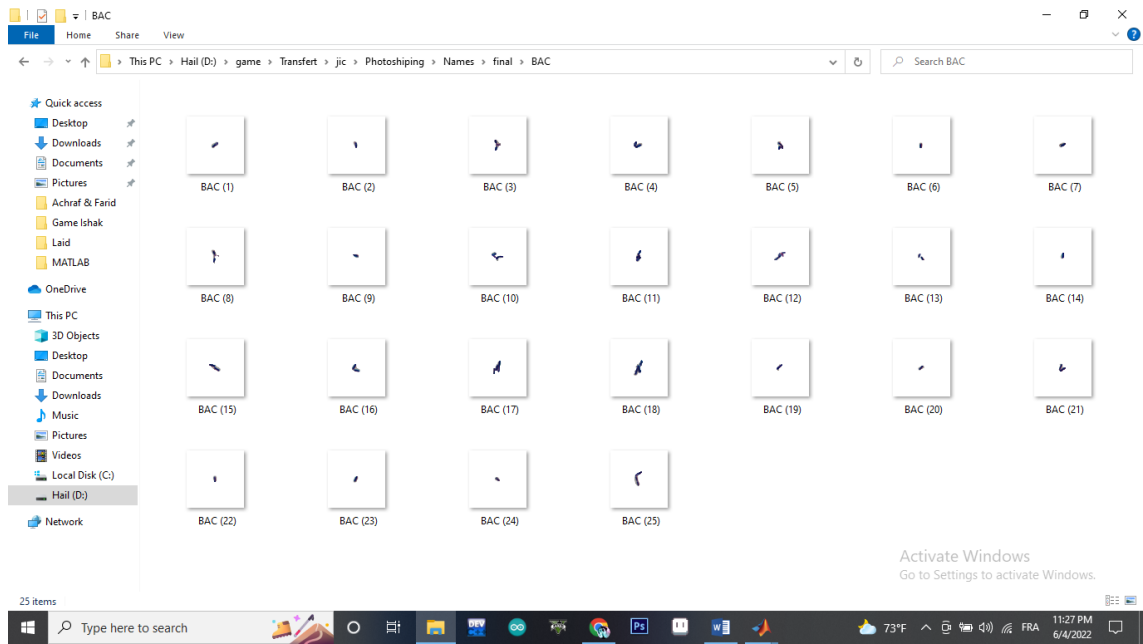

Figure 3.5: Hippuric Acid base images that were used in image generation

Figure 3.6: Bacterial base images that were used in image generation
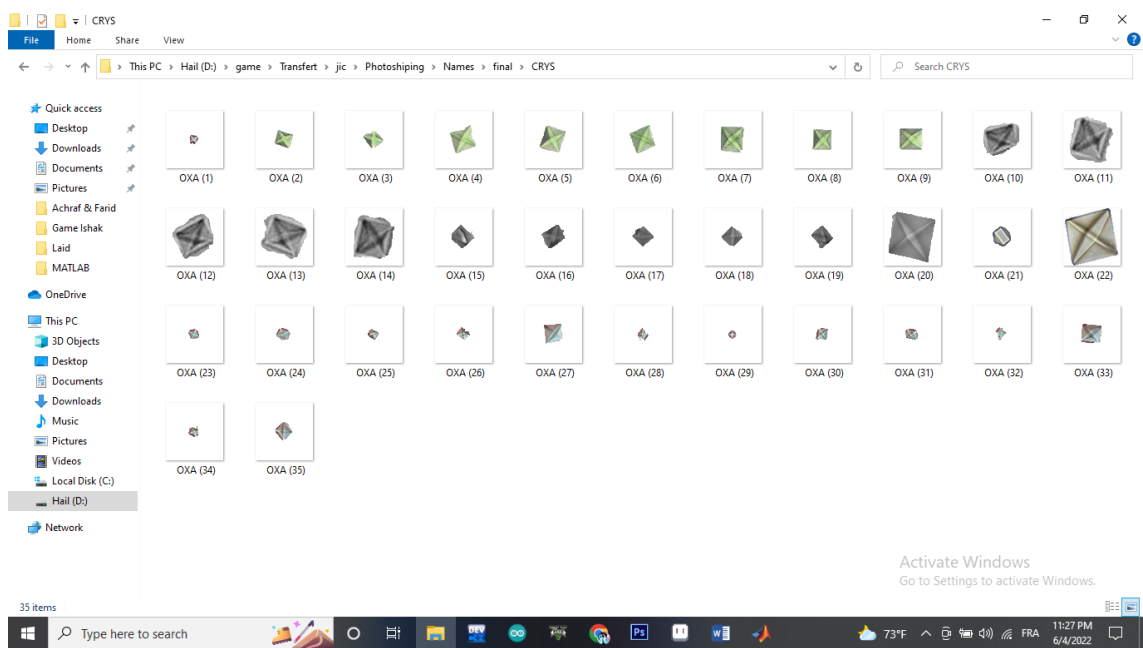
Figure 3.7: Oxalate Calcium base images that were used in image generation

51
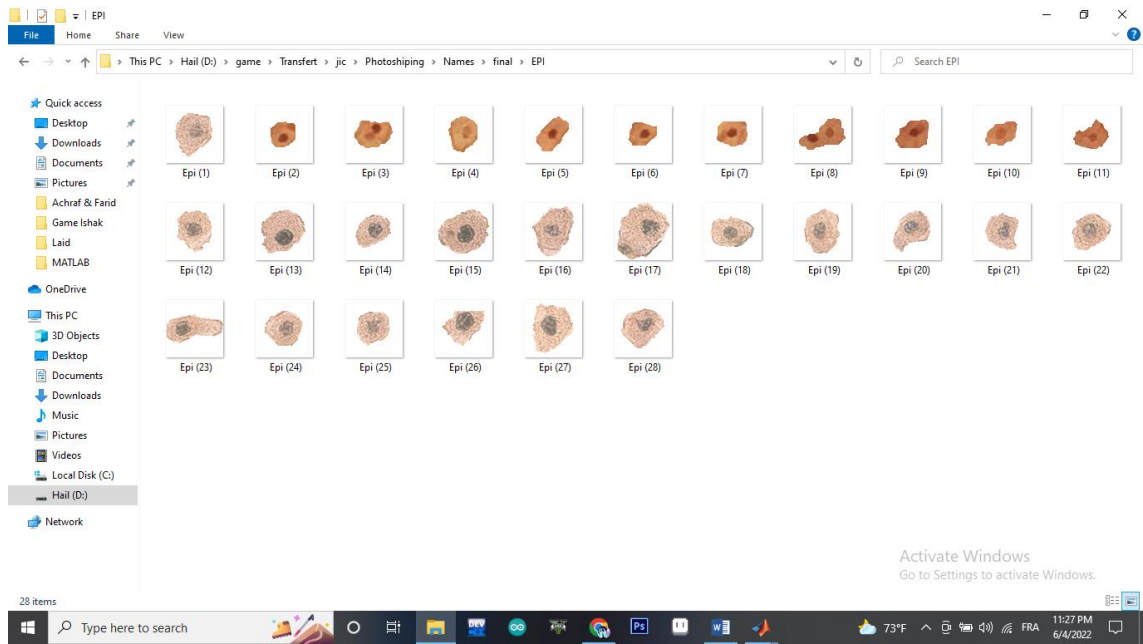
Figure 3.8: Epithilial base images that were used in image generation
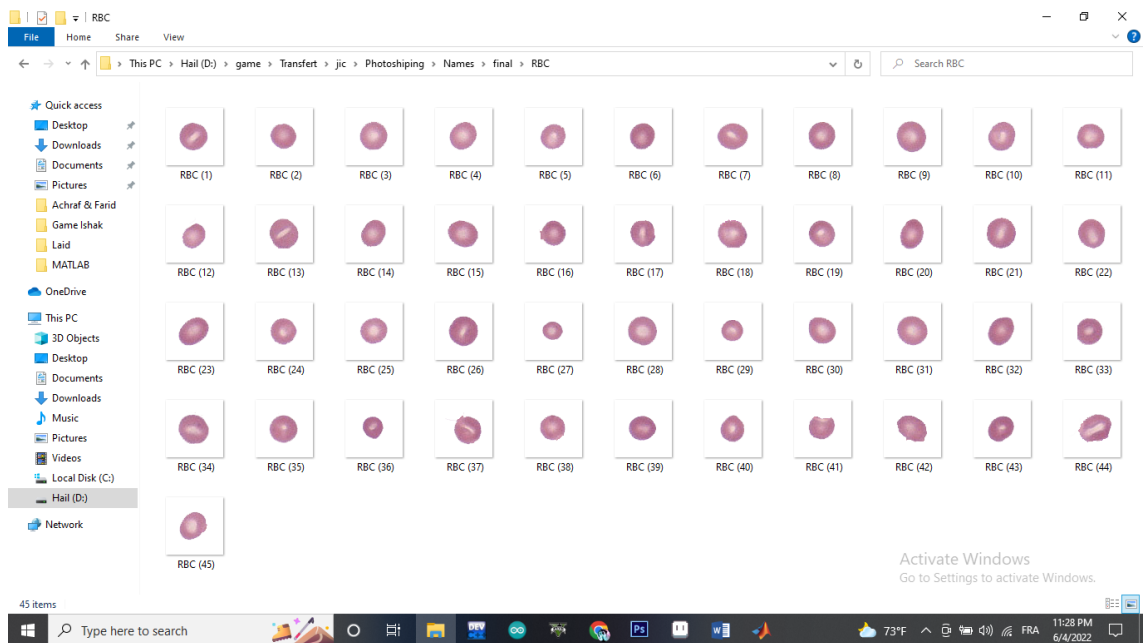
Figure 3.9: Red Blood Cells base images that were used in image generation
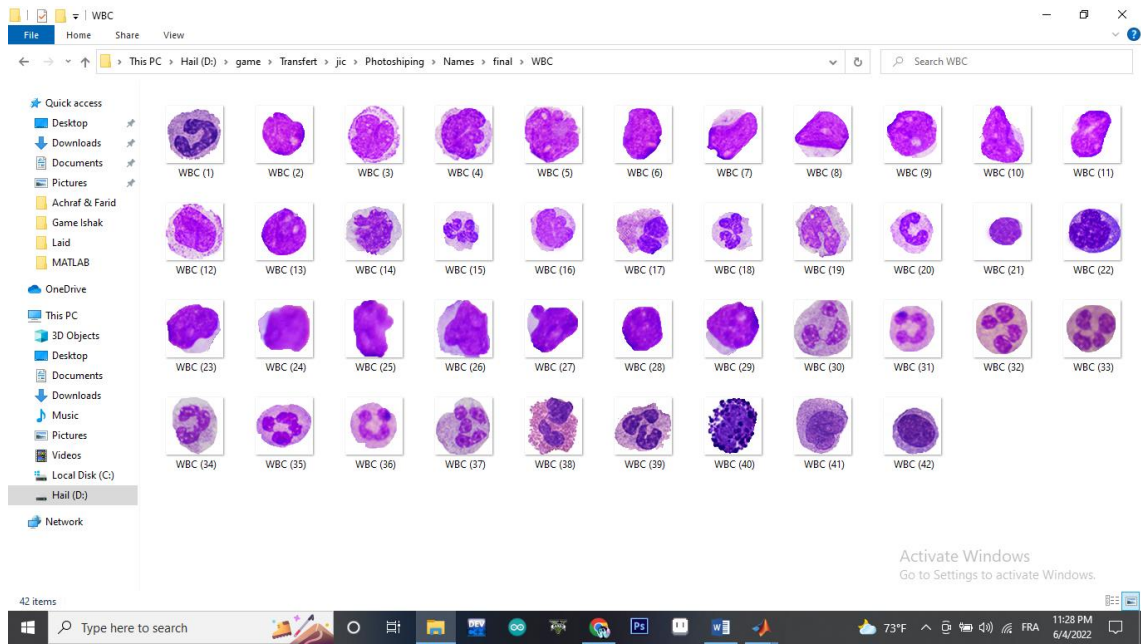
Figure 3.10: White Blood Cells base images that were used in image generation
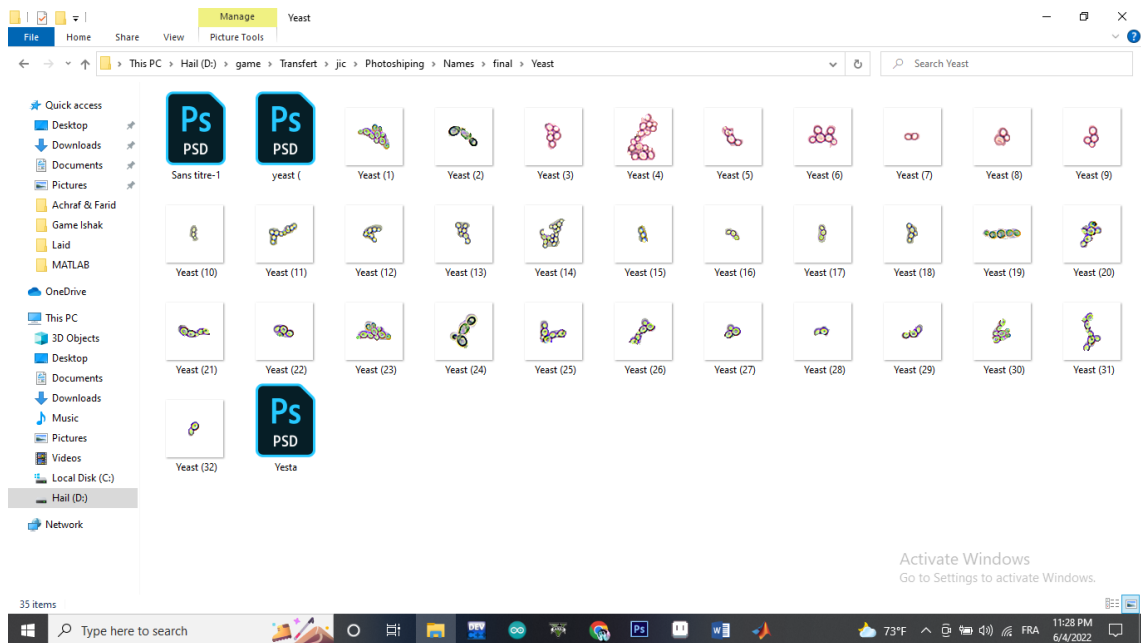
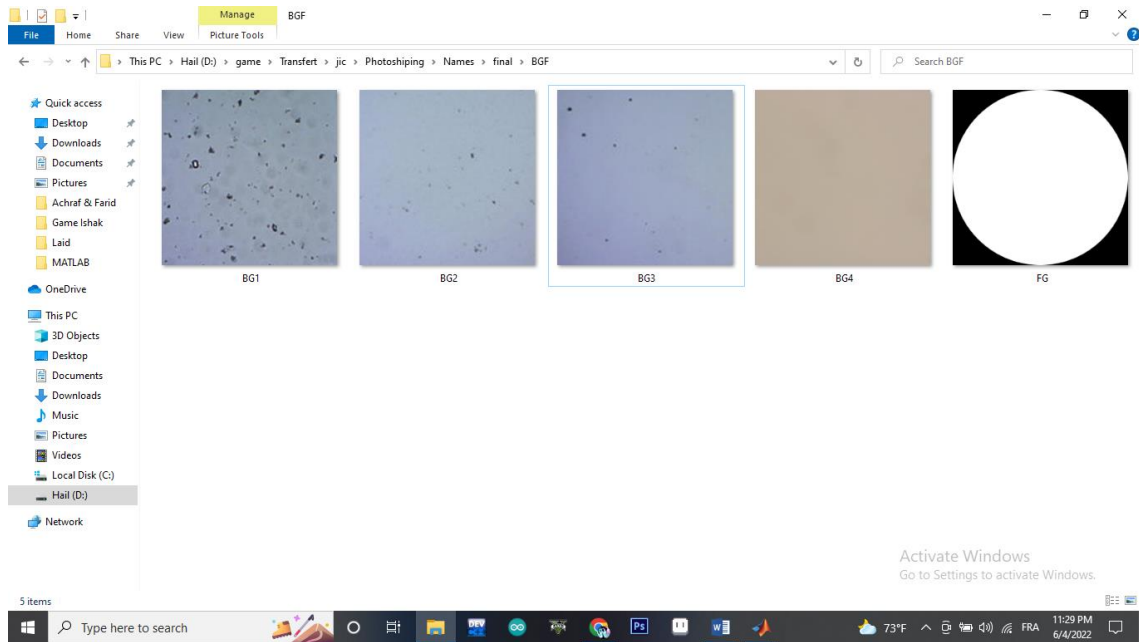Figure 3.11: Yeast base images that were used in image generation

Figure 3.12: Background and Foreground base images that were used in image generation

3.3.2 MATLAB Image Generator:

To simplify the work for each type of pathogens we created a 100 images that contains only that type of pathogen except for White Blood cells, where we added some Red Blood cells to it.

We start by loading the pathogen images into the memory then choosing a random number each time that represent the number of cells that we will generate in each image inside a loop were we generate other random variables that plays the role of coordinates and opacity level for realism purposes.

Before adding the cells we first load the background images then add on top of it after the loop ends we add a black picture with a transparent circle shape on the middle to make it similar to the microscopic view.

Annotations which are a crucial part of this project are added simultaneously and this latter consist of the X and Y coordinates of the cell plus the Width and Height of the bounding box around it this will be later saved into a txt file for each image, the yolo family uses this annotation format as follow:

 <object class> <X> <Y> <Width> <Height>.

Where object class can be an integer and the others as a number between 0 and 1.

After each iteration an image will be saved as a png with a txt file that has the same name, but before the saving process we added some randomized noise and blur to make the image looks a bit realistic.
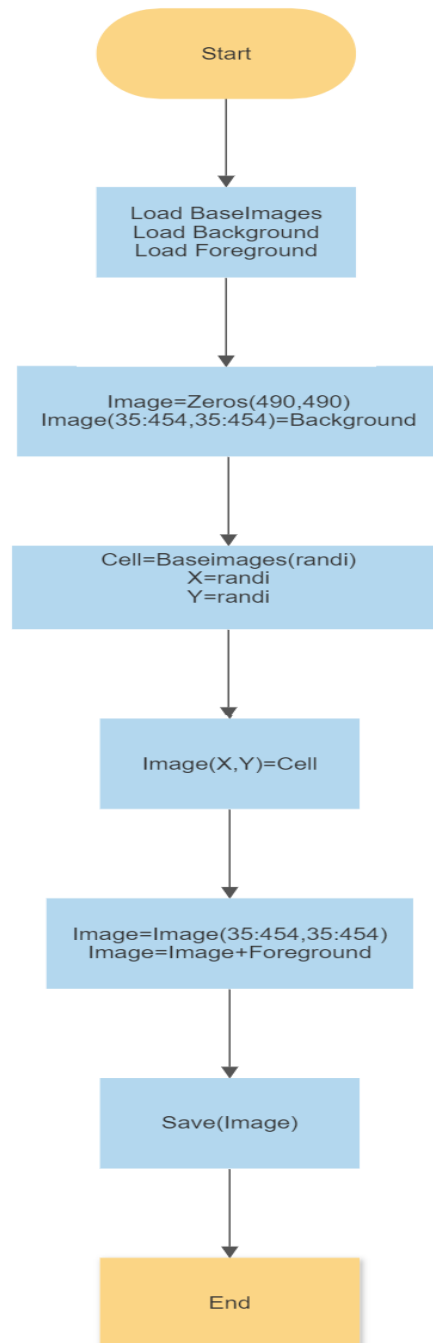
The whole process is presented in this flow-chart:

Figure 3.13: FlowChart of Pseudo Microscopic Generator
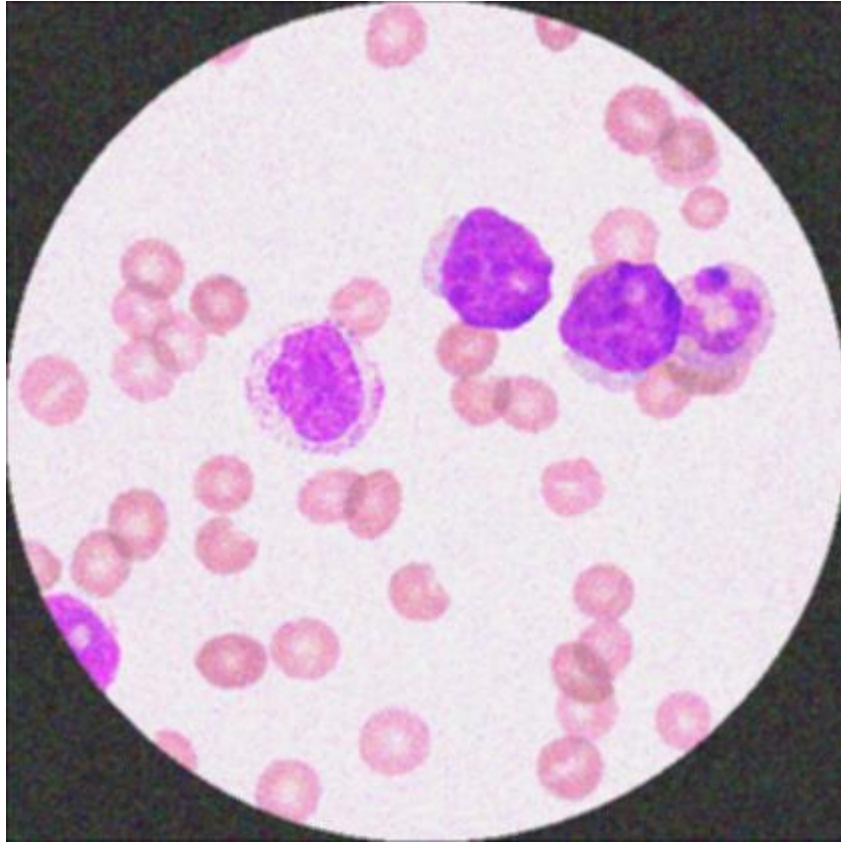
The end result would look like:

Figure 3.14: Pseudo Microscopic Image of White and Red Blood Cells
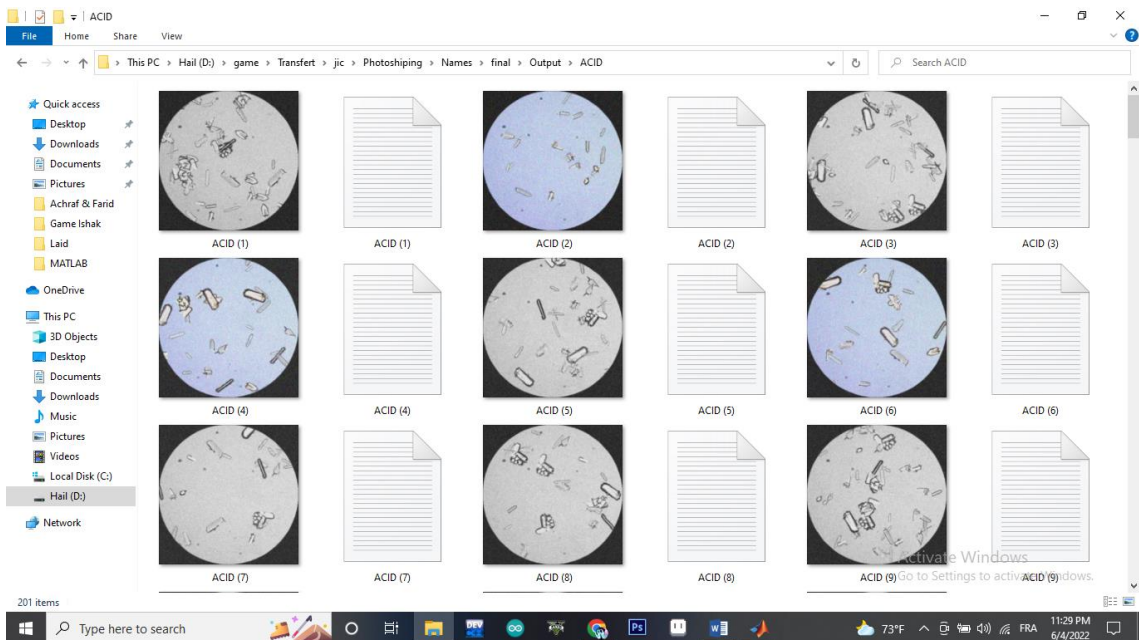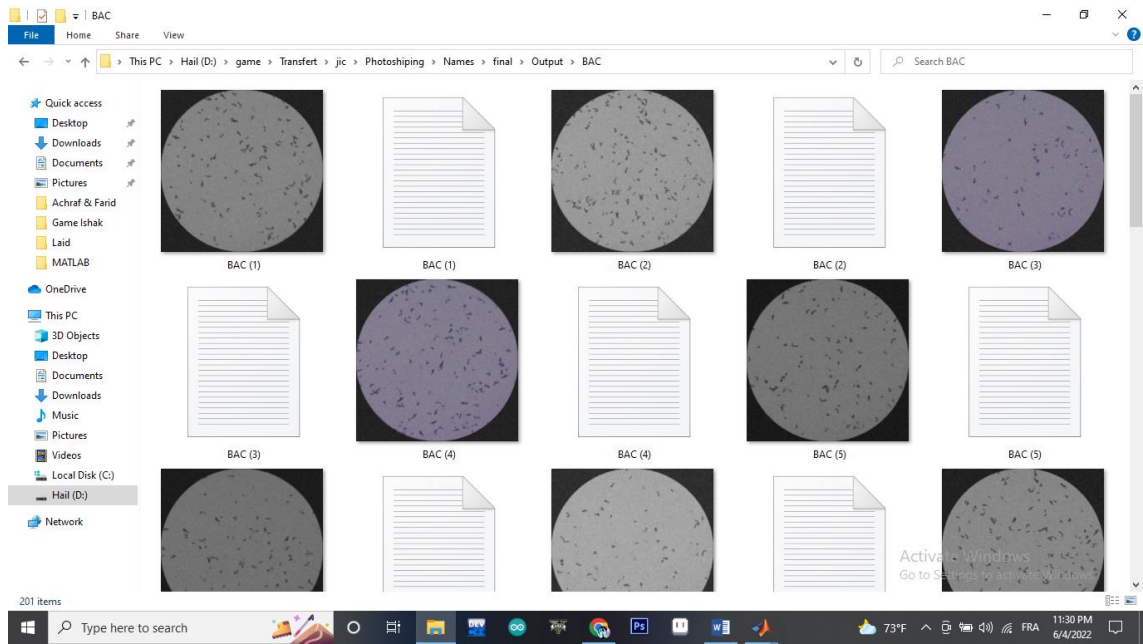

Figure 3.15: Output of Hippuric Acid images with Annotation file

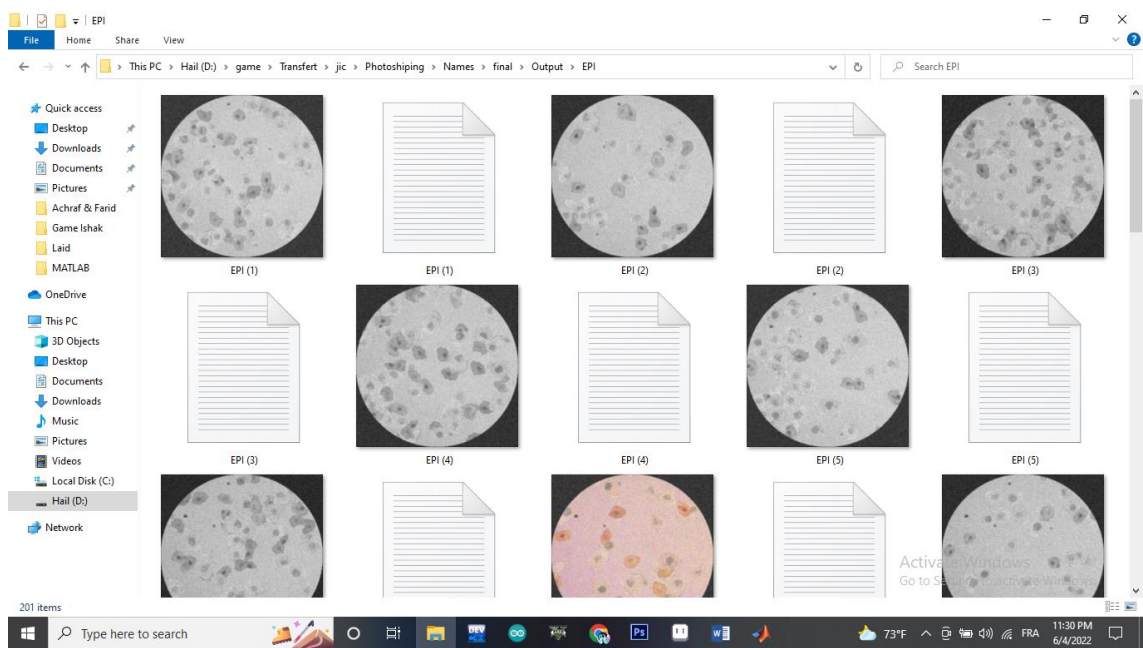Figure 3.16: Output of Bacterial images with Annotation file

Figure 3.17: Output of Epithilial Cells images with Annotation file
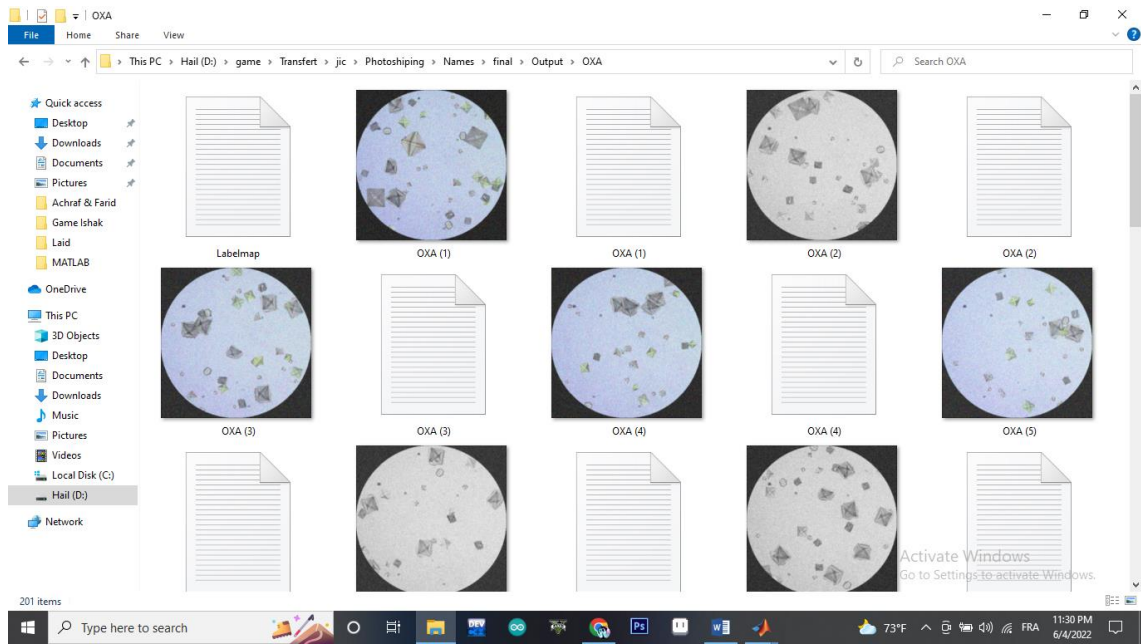
Figure 3.18: Output of Oxalate Calcium images with Annotation file
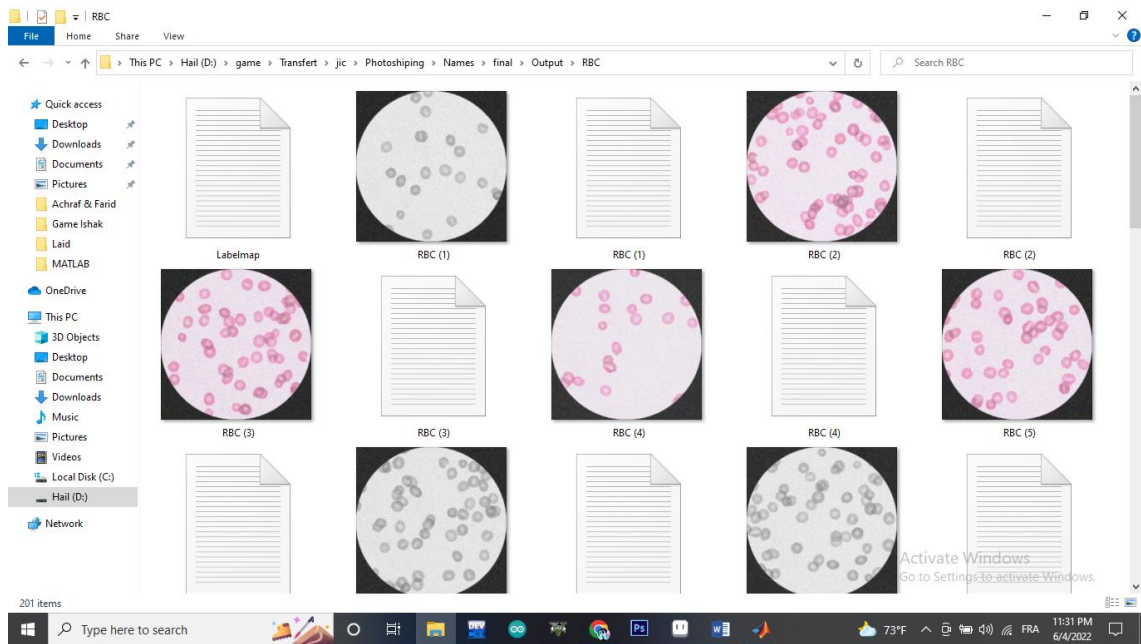

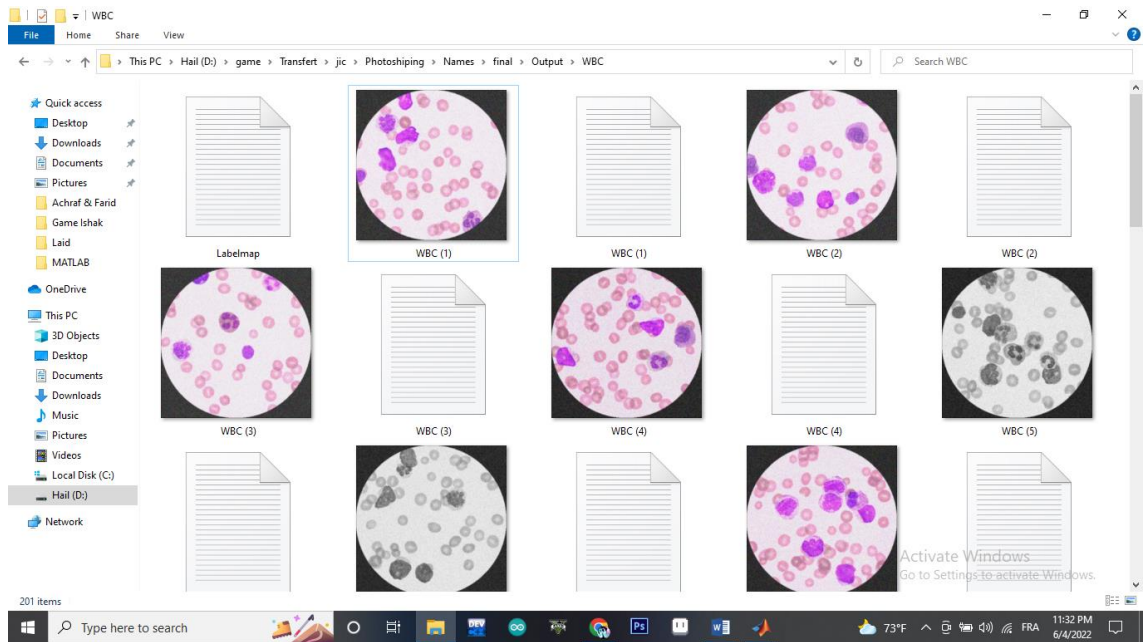Figure 3.19: Output of Red Blood Cells images with Annotation file

Figure 3.20: Output of Red & White Blood Cells images with Annotation file

All this process it would take few minutes to generate all the 700 images that later was uploaded to an automated service for augmentation which is called Roboflow.

3.4 Training and results:

In this section we will discuss the steps taken to prepare and train the model and finaly discussing the results.

3.4.1 Training phase:

The training phase consists of loading the colab with datasets and Yolo v5 repository from github.

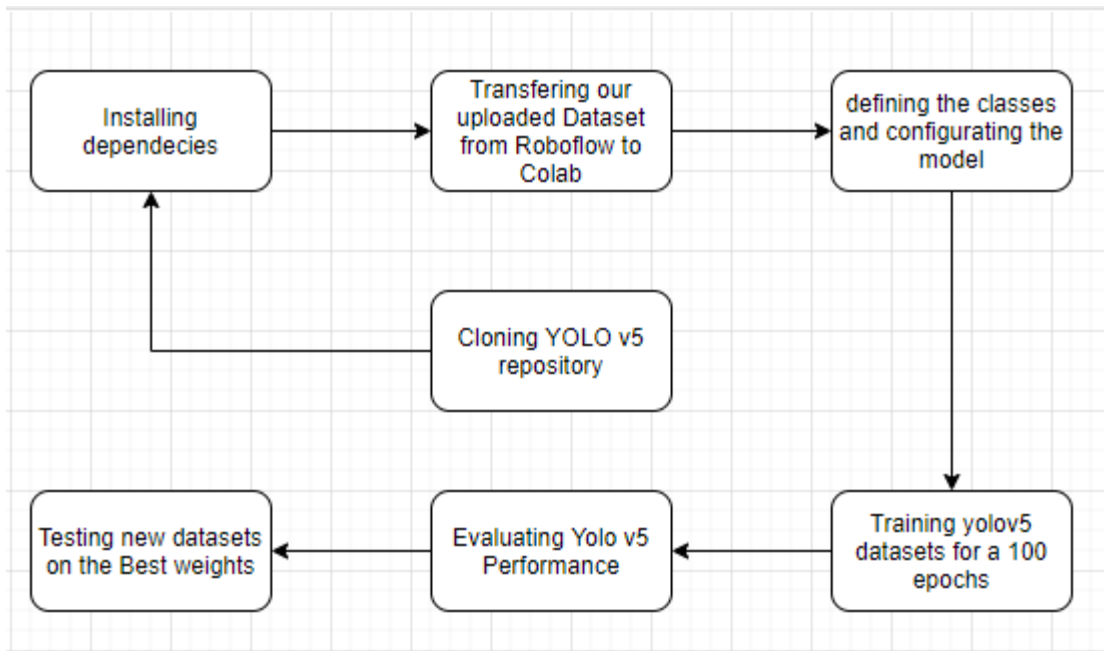The following flowchart will explain how the steps are executed:

Figure 3.21: flowchart of steps of training and evaluation

First we clone the Yolo v5 repository from the github then we install all the dependencies, later we transfer the augmented dataset that we've uploaded before on the roboflow service into the google colab . Yolo v5 uses a txt annotation system and also w file directory system called YAML that defines the directories of each images which was generated by roboflow.

Labelmap is a file that we created alongside the dataset that we uploaded it contains all the translations of numerical annotations into a human language annotation.

For the configuration part there are three area that yolo needs to be configured:

- Anchors: are a set of predefined bounding boxes of a certain height and width. These boxes are defined to capture the scale and aspect ratio of specific object classes you want to detect and are typically chosen based on object sizes in your training datasets.
- Backbone: A convolutional neural network that aggregates and forms image features at different granularities.
- Head : The role of the head in the case of a one stage detector is to perform dense prediction. The dense prediction is the final prediction which is composed of a vector containing the coordinates of the predicted bounding box (center, height, width), the confidence score of the prediction and the label.

After setting up all of the above we started the training for a 100 epochs so we can get the most out of our dataset and it took 1 hour and 5 minutes.
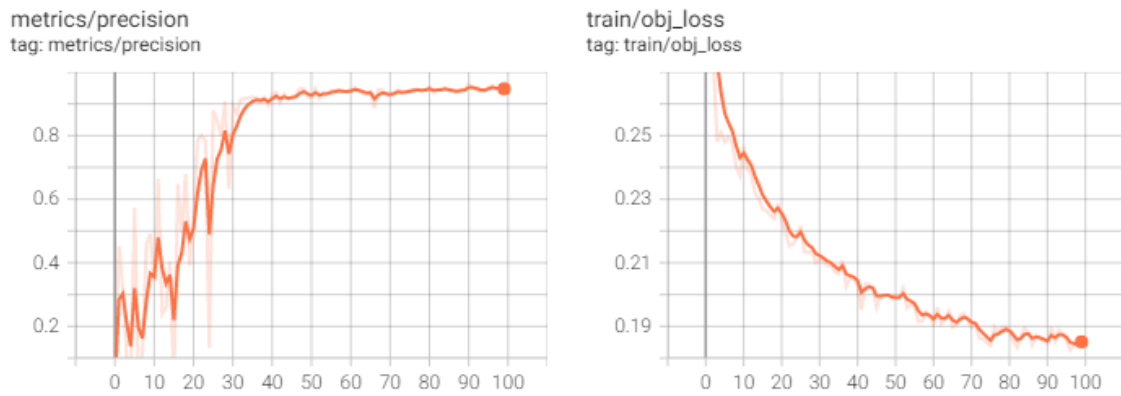
3.4.2 Evaluating the results:

Figure 3.22:plot that shows the precision and the loss

As we can see from the plot of (figure 3.4) on the metrics/precision we've already reached our most accuracy after approximately 50 epochs.

After the training finished we ended up with two types of weights one is at the best accuracy and one at the end of all epochs then we tested non pseudo microscopic images on the best trained weights to see the results.
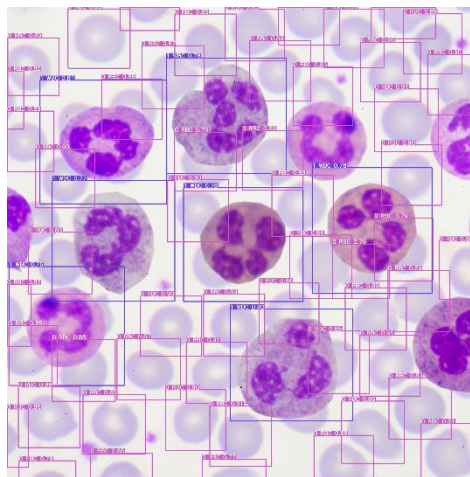

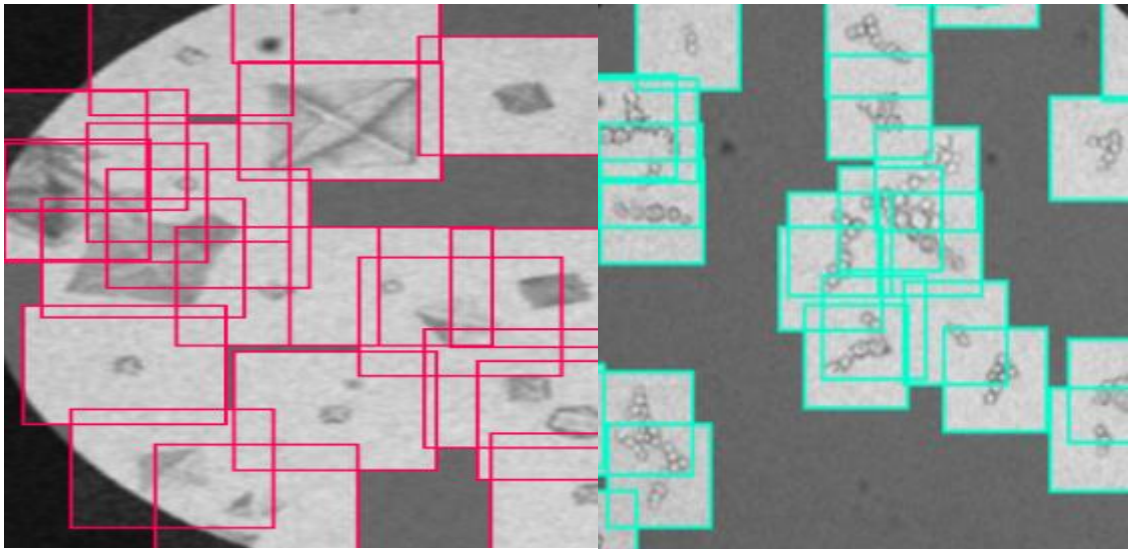Figure 3.23: Real microscopic image of Blood cells detected by Yolo v5

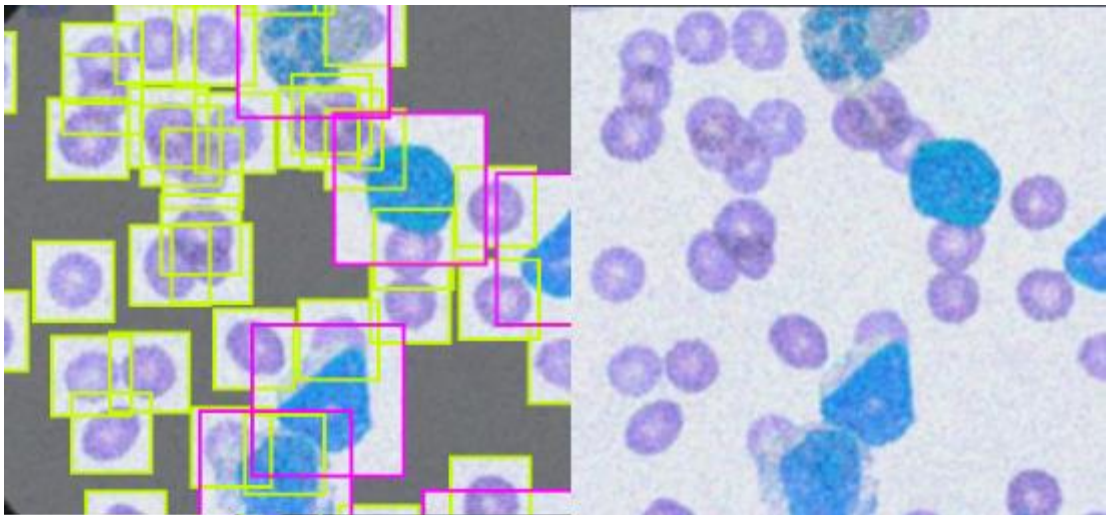Figure 3.24: Oxalate Calcium and Yeast detection on Gray scale

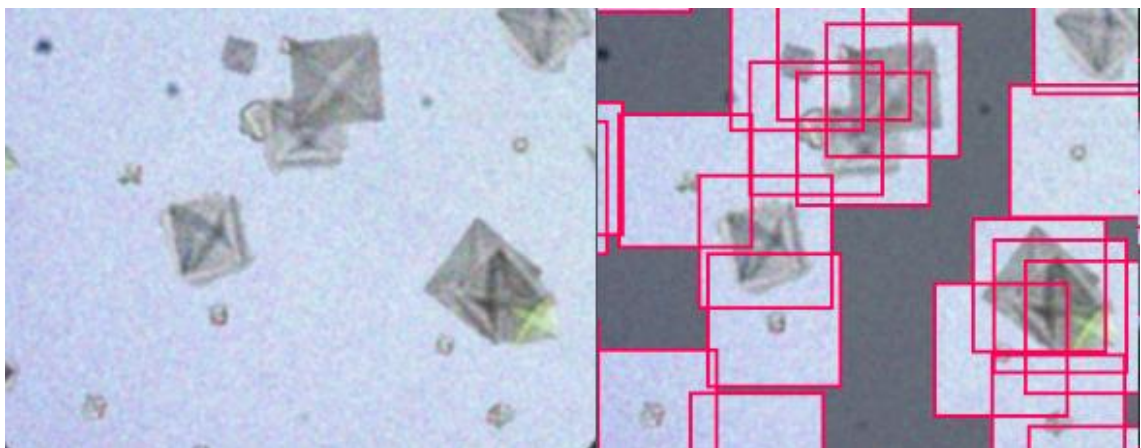Figure 3.25: White and Red Blood cells detection after color shifting

Figure 3.26: Oxalate Calcium before and after Detection

3.5 Conclusion:

In this chapter we've shown how we made our MATLAB program that generates pseudo microscopic images from standalone pathogens which later we augmented and used to train our Yolo v5 model and had a satisfying results.

We can have better results by using more standalone pathogens to generate a big variety of pseudo images and also using more developed technics of image modifications, we can also train each pathogenic type on its own Yolo v5 model while changing the anchors that suits the detected object

General Conclusion

In this project we wanted to achieve a simplified pathogens detection and classification under a microscope due to the lack of expensive equipment in Algerian Hospitals and private low cash flow laboratories.

So we created a datasets through a MATLAB code that generates Pseudo microscopic images alongside with their annotations as bounding boxes using standalone pathogens with transparent background, that later was augmented to have a variety of images that we can use for training.

We trained the latter in YOLO v5 model on a Google Colab for a 100 epochs that took only 1h and 5 minutes that ended in a satisfying results.

There is always a room for improvements by adding more variety of standalone pathogens and using different ways of image deformations, and also remodeling the anchors of the Yolo model for each pathogen depend on its size, all of the above will help obtain better and more accurate results.

The weights obtained can be later transformed into an application that runs in a laptop or an android device.

Bibliographie

[1] https://www.microscopemaster.com/bacteria-size-shape-arrangement.html

[2] https://biologyreader.com/bacterial-shape-and-arrangement.html

[3] Masaya Suzuki, Yosuke Asada, Daisuke Watanabe, Daisuke Watanabe: Cell shape and growth of budding yeast cells in restrictive microenvironments , 2004.

[4]https://www.khanacademy.org/science/biology/biology-of-viruses/virus-biology/a/intro-to-viruses

[5] https://trialsjournal.biomedcentral.com/articles/10.1186/s13063-019-3914-2

[6] https://www.diabetes.co.uk/body/white-blood-cells.html

[7] https://www.britannica.com/science/red-blood-cell

[8] https://askabiologist.asu.edu/epithelial-cells

[9] https://www.degruyter.com/document/doi/10.1515/cclm-2015-0480/html

[10] https://www.medicalnewstoday.com/articles/urine-crystals#types

[11] https://www.sciencedirect.com/topics/medicine-and-dentistry/leukocyturia

[12]https://www.glowm.com/lab-text/item/94#:~:text=Urine%20is%20normally%20sterile.,ml%20represents%20urinary%20tract%20infection.

[13]https://my.clevelandclinic.org/health/diseases/16854-cerebrospinal-fluid-csf-leak#:~:text=Cerebrospinal%20fluid%20(CSF)%20is%20a,removal%20system%20for%20the%20brain.

[14]https://www.ucsfhealth.org/medical-tests/wbc-count#:~:text=The%20normal%20number%20of%20WBCs,or%20may%20test%20different%20specimens.

[15] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3854183/

[16] https://pubmed.ncbi.nlm.nih.gov/15307030/

[17]https://www.sciencedirect.com/topics/earth-and-planetary-sciences/artificial-neural-network

[18] https://www.simplilearn.com/tutorials/deep-learning-tutorial/perceptron

[19] https://deepai.org/machine-learning-glossary-and-terms/multilayer-perceptron

[20] https://www.mathworks.com/discovery/deep-learning.html

[21]https://towardsdatascience.com/machine-learning-fundamentals-via-linear-regression-41a5d11f5220

[22] https://en.wikipedia.org/wiki/Unsupervised_learning

[23]https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/

[24]https://www.analyticsvidhya.com/blog/2020/10/what-is-the-convolutional-neural-network-architecture/

[25]https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/#:~:text=Convolutional%20layers%20are%20the%20major,that%20results%20%20in%20an%20activation.&text=The%20result%20is%20highly%20specific,detected%2%200anywhere%20on%20input%20images.

[26]https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

[27]https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/#:~:text=Pooling%20layers%20are%20used%20to,generated%20by%20a%20convolution%20layer

[28] https://iq.opengenus.org/fully-connected-layer/

[29] https://neurohive.io/en/popular-networks/vgg16/

[30] https://pjreddie.com/darknet/yolo/

[31] Colab.reaserch.google.com

[32] https://research.google.com/colaboratory/faq.html?hl=fr

[33] https://blog.education-ecosystem.com/what-is-google-colab

[34] Introduction au logiciel Matlab Marie Postel∗ Version r´evis´ee septembre 2004

[35]https://medium.com/analytics-vidhya/object-detection-algorithm-yolo-v5-architecture-89e0a35472ef

[36] https://github.com/ultralytics/yolov5

[37] https://medium.com/deelvin-machine-learning/yolov4-vs-yolov5-db1e0ac7962b