





# UNIVERSIDAD DE INVESTIGACIÓN DE TECNOLOGÍA EXPERIMENTAL YACHAY

Escuela de Ciencias Matemáticas y Computacionales

## TÍTULO: AUTOMATIC HIGH-RESOLUTION ANALYSIS OF PAP TEST CELLS

Trabajo de integración curricular presentado como requisito para la obtención del título  
de Ingeniero en Tecnologías de la Información

### **Autor**

Toapanta Maila Bryan Oswaldo

### **Tutor**

Ph.D. Chang Tortolero Oscar Guillermo

Urucuquí - Mayo 2021

**SECRETARÍA GENERAL**  
**(Vicerrectorado Académico/Cancillería)**  
**ESCUELA DE CIENCIAS MATEMÁTICAS Y COMPUTACIONALES**  
**CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN**  
**ACTA DE DEFENSA No. UITEY-ITE-2021-00009-AD**

A los 26 días del mes de mayo de 2021, a las 16:00 horas, de manera virtual mediante videoconferencia, y ante el Tribunal Calificador, integrado por los docentes:

<b>Presidente Tribunal de Defensa</b>	Dr. CAMACHO , FRANKLIN , Ph.D.
<b>Miembro No Tutor</b>	Dr. ANTON CASTRO , FRANCESC , Ph.D.
<b>Tutor</b>	Dr. CHANG TORTOLERO, OSCAR GUILLERMO , Ph.D.

El(la) señor(ita) estudiante **TOAPANTA MAILA, BRYAN OSWALDO**, con cédula de identidad No. **1718587205**, de la **ESCUELA DE CIENCIAS MATEMÁTICAS Y COMPUTACIONALES**, de la Carrera de **TECNOLOGÍAS DE LA INFORMACIÓN**, aprobada por el Consejo de Educación Superior (CES), mediante Resolución **RPC-SO-43-No.496-2014**, realiza a través de videoconferencia, la sustentación de su trabajo de titulación denominado: **Automatic High Resolution Analysis of Pap Test Cells** , previa a la obtención del título de **INGENIERO/A EN TECNOLOGÍAS DE LA INFORMACIÓN**.

El citado trabajo de titulación, fue debidamente aprobado por el(los) docente(s):

<b>Tutor</b>	Dr. CHANG TORTOLERO, OSCAR GUILLERMO , Ph.D.
--------------	--

Y recibió las observaciones de los otros miembros del Tribunal Calificador, las mismas que han sido incorporadas por el(la) estudiante.

Previamente cumplidos los requisitos legales y reglamentarios, el trabajo de titulación fue sustentado por el(la) estudiante y examinado por los miembros del Tribunal Calificador. Escuchada la sustentación del trabajo de titulación a través de videoconferencia, que integró la exposición de el(la) estudiante sobre el contenido de la misma y las preguntas formuladas por los miembros del Tribunal, se califica la sustentación del trabajo de titulación con las siguientes calificaciones:

Tipo	Docente	Calificación
Tutor	Dr. CHANG TORTOLERO, OSCAR GUILLERMO , Ph.D.	10,0
Presidente Tribunal De Defensa	Dr. CAMACHO , FRANKLIN , Ph.D.	10,0
Miembro Tribunal De Defensa	Dr. ANTON CASTRO , FRANCESC , Ph.D.	10,0

Lo que da un promedio de: **10 (Diez punto Cero)**, sobre 10 (diez), equivalente a: **APROBADO**

Para constancia de lo actuado, firman los miembros del Tribunal Calificador, el/la estudiante y el/la secretario ad-hoc.

**TOAPANTA MAILA, BRYAN OSWALDO**  
**Estudiante**

BRYAN OSWALDO TOAPANTA MAILA  
 Firmado digitalmente por BRYAN OSWALDO TOAPANTA MAILA  
 Fecha: 2021.06.10 19:10:38 -05'00'

Dr. CAMACHO , FRANKLIN , Ph.D.  
**Presidente Tribunal de Defensa**

Firmado electrónicamente por:  
**FRANKLIN JOSE**

Dr. CHANG TORTOLERO, OSCAR GUILLERMO , Ph.D.  
**Tutor**

Firmado electrónicamente por:  
**OSCAR GUILLERMO CHANG TORTOLERO**

**FRANCESC  
ANTON  
CASTRO**

Signé électroniquement par  
FRANCESC ANTON CASTRO  
cn=FRANCESC ANTON CASTRO, ou=  
ENTIDAD DE CERTIFICACION DE  
INFORMACION, o= SECURITY DATA S.  
A. 1, c= EC  
Date: 2021.06.08 13:36:37 ECT

Dr. ANTON CASTRO , FRANCESC , Ph.D.  
**Miembro No Tutor**

DAYSY MARGARITA MEDINA BRITO  Firmado digitalmente por DAYSY  
MARGARITA MEDINA BRITO  
Fecha: 2021.06.08 09:07:46 -05'00'

MEDINA BRITO, DAYSY MARGARITA  
**Secretario Ad-hoc**

# Autoría

Yo, **Bryan Oswaldo Toapanta Maila**, con cédula de identidad **1718587205**, declaro que las ideas, juicios, valoraciones, interpretaciones, consultas bibliográficas, definiciones y conceptualizaciones expuestas en el presente trabajo; así cómo, los procedimientos y herramientas utilizadas en la investigación, son de absoluta responsabilidad de el autor del trabajo de integración curricular. Así mismo, me acojo a los reglamentos internos de la Universidad de Investigación de Tecnología Experimental Yachay.

Urcuquí, Junio del 2021.

---

Bryan Oswaldo Toapanta Maila  
CI: 1718587205

# Autorización de publicación

Yo, **Bryan Oswaldo Toapanta Maila**, con cédula de identidad **1718587205**, cedo a la Universidad de Tecnología Experimental Yachay, los derechos de publicación de la presente obra, sin que deba haber un reconocimiento económico por este concepto. Declaro además que el texto del presente trabajo de titulación no podrá ser cedido a ninguna empresa editorial para su publicación u otros fines, sin contar previamente con la autorización escrita de la Universidad.

Asimismo, autorizo a la Universidad que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Urcuquí, Junio del 2021.

---

Bryan Oswaldo Toapanta Maila  
CI: 1718587205

# Dedication

*To my family, Olga, Oswaldo, and Ayme, who always supported me in every way,  
especially with their love and affection.*

*To all my friends without whom I would have gone crazy with stress years ago.*

*Oswaldo Toapanta Mailla*

# Acknowledgments

Thanks to my parents Olga and Oswaldo, who constantly inspire me to keep working. Thanks to my sister Ayme who, together with my parents, always encouraged me to finish this and every assignment that led me to this dissertation. In general, thanks to every family member who believed in me.

I want to express my gratitude to my advisor, Ph.D. Oscar Chang, for all the patience and time he invested in me during this project's development. Also, I want to thank every teacher involved in my education, who did not limit themselves only to teach but to inspire their students.

Finally, I want to thank all my special friends. To my high school friends, Johanna, Paola, and Karen, who are always there to help me and support me despite the distance. To all my college friends, specially to "la Casa 14", who made of the last five years an incredible experience; I would love to list you all but you guys are too many and you know who you all are. I hope someday I can believe in myself as much as you guys believe in me.

*Oswaldo Toapanta Maila*

# Resumen

Según la Organización Mundial de la Salud (OMS), el cuarto tipo de cáncer más frecuente en mujeres es el de cuello uterino. En Ecuador en 2018, hubo 1612 casos de esta enfermedad y 838 muertes. Una práctica vital para evitar estas pérdidas es el diagnóstico temprano de la enfermedad, que se logra con la prueba de Papanicolaou. Una de las labores del médico o técnico patólogo es revisar la muestra de Papanicolaou y determinar si el paciente tiene células cancerosas sospechosas o evidentes. En Ecuador, el médico revisa el frotis de Papanicolaou “manualmente”, es decir, tiene que revisar la imagen del microscopio digital del frotis poco a poco hasta cubrir toda la imagen; esta búsqueda puede llevar hasta 30 minutos para un operador experimentado. En 2014, había menos de 389 patólogos en Ecuador, lo que muestra una falta de personal y, en consecuencia, una dificultad importante para el diagnóstico temprano. Esta tesis utiliza redes neuronales convolucionales en el ambiente de TensorFlow para crear un sistema de escaneo rápido para el reconocimiento de células. El método propuesto emplea un proceso de dos etapas. La primera etapa es un escaneo de alta velocidad de baja resolución que detecta las células, las amplía y luego pasa una versión de alta resolución a la siguiente etapa. La segunda etapa se encarga de la clasificación de alta resolución. La máquina se creó utilizando un conjunto de 963 imágenes de frotis de Papanicolaou de base líquida. Estas imágenes corresponden a 460 pacientes y fueron tomadas con un aumento de 40x utilizando un microscopio Leica ICC50 HD. El software ya entrenado clasifica las células en normales y anormales para que luego, un patólogo altamente capacitado pueda revisar las muestras sospechosas de Papanicolaou haciendo un análisis de cerca, ayudando a los patólogos a lograr un diagnóstico más temprano y mejorando el servicio médico.

**Palabras Clave:** redes neuronales, papanicolaou, células, cancer de cuello uterino.

# Abstract

According to the World Health Organization (WHO), the fourth most frequent cancer in women is cervix cancer. Specifically for Ecuador in 2018, there were 1612 cases of this disease and 838 deaths. A vital practice to avoid these losses is an early diagnosis of the disease, accomplished by the Papanicolaou or Pap test. One of the doctor or technician pathologists' labors is to check the so-called pap smear and determine if the corresponding patient has suspicious or evident cancer cells. In Ecuador, the doctor checks pap smear "manually," that is, they have to check the digital microscope picture of the smear little by little until the whole picture is covered; this search may take up to 30 minutes for an experienced operator. By 2014, there were less than 389 pathologists in Ecuador, showing a lack of personnel and, in consequence, a severe difficulty for early diagnosis. This thesis uses convolutional neural networks in the TensorFlow ambient to create a fast scanning abnormal cell recognition system. The proposed method employs a two-stage process. The first stage is a low-resolution high-speed scanning that detects cells, zooms into them, and then passes a high-resolution version to the next stage. The second stage is in charge of the high-resolution classification. The software was created using a set of 963 liquid-based pap smear images. These images corresponded to 460 patients and were taken with a 40x magnification using a Leica ICC50 HD microscope. The trained system classifies cells into normal and abnormal so that later, a highly trained pathologist can review suspicious pap smear samples doing a close-up analysis, helping pathologists achieve earlier diagnosis and improving medical service.

**Keywords:** neural network, pap smear, cell, cervix cancer, screen.

# Contents

<b>Contents</b>	<b>i</b>
<b>List of Tables</b>	<b>iii</b>
<b>List of Figures</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Objectives . . . . .	2
1.1.1 General Objective . . . . .	2
1.1.2 Specific Objectives . . . . .	2
<b>2 Problem Statement</b>	<b>3</b>
<b>3 Related Work</b>	<b>6</b>
<b>4 Technical Background</b>	<b>8</b>
4.1 Neural Networks . . . . .	8
4.1.1 Hyperparameters . . . . .	10
4.1.2 Parameters . . . . .	10
4.1.3 Activation Function . . . . .	11
4.1.4 Loss Function . . . . .	13
4.1.5 Gradient Descent . . . . .	14
4.1.6 Backpropagation . . . . .	15
4.1.7 Underfitting and Overfitting . . . . .	15
4.2 Interconnection . . . . .	16
4.2.1 Feed Forward Connection . . . . .	16
4.2.2 Recurrent Network . . . . .	17
4.3 Learning Rule . . . . .	17
4.3.1 Supervised Learning . . . . .	17
4.3.2 Unsupervised Learning . . . . .	18
4.3.3 Reinforcement learning . . . . .	18
4.4 Convolutional Networks . . . . .	18
4.4.1 Filter . . . . .	19
4.4.2 Convolution Operation . . . . .	19
4.4.3 Padding . . . . .	19

4.4.4	Strides . . . . .	20
4.4.5	Pooling . . . . .	21
4.4.6	Dropout . . . . .	21
4.5	Data Augmentation . . . . .	22
4.6	Confusion Matrix . . . . .	22
4.6.1	Confusion Metrics . . . . .	23
<b>5</b>	<b>Data Description</b>	<b>24</b>
<b>6</b>	<b>Implementation</b>	<b>26</b>
6.1	Technology . . . . .	26
6.1.1	Software . . . . .	26
6.1.2	Hardware . . . . .	27
6.2	Parameters Setting . . . . .	27
6.2.1	Region of Interest . . . . .	27
6.2.2	Batch size . . . . .	28
6.2.3	Epoch . . . . .	28
6.2.4	Activation Function . . . . .	28
6.2.5	Dropout . . . . .	28
6.2.6	Maxpooling . . . . .	28
6.2.7	Filters . . . . .	29
6.2.8	Strides and Padding . . . . .	29
6.3	Cell recognition Model Implementation . . . . .	29
6.3.1	Data Preparation . . . . .	29
6.3.2	Convolutional Neural Network Architecture . . . . .	31
6.3.3	Training . . . . .	32
6.4	Cell classification model implementation . . . . .	32
6.4.1	Data Preparation . . . . .	32
6.4.2	Convolutional Neural Network Architecture . . . . .	34
6.4.3	Training . . . . .	37
6.5	Scanning Process . . . . .	37
<b>7</b>	<b>Results</b>	<b>40</b>
<b>8</b>	<b>Conclusions</b>	<b>44</b>
	<b>Bibliography</b>	<b>46</b>
	<b>Appendices</b>	<b>53</b>

# List of Tables

4.1	Cell Classification Model Results . . . . .	22
7.1	Cell Recognition Model Results . . . . .	41
7.2	Cell Classification Model Results . . . . .	41

# List of Figures

4.1	Biological Neuron. . . . .	8
4.2	Neural Network Architecture. . . . .	9
4.3	Binary Step Activation. . . . .	11
4.4	Linear Activation Function. . . . .	11
4.5	Sigmoid Activation Function. . . . .	12
4.6	RELU Activation Function. . . . .	12
4.7	Softmax Activation Function. . . . .	13
4.8	Feed Forward Connection. . . . .	16
4.9	Recurrent Network. . . . .	17
4.10	Filter . . . . .	19
4.11	Convolution Example . . . . .	19
4.12	Padding Example . . . . .	20
4.13	Stride of 3 . . . . .	20
4.14	Max pooling vs Average Pooling . . . . .	21
4.15	Standard vs Dropout . . . . .	21
4.16	Data augmentation . . . . .	22
5.1	Negative for Intraepithelial lesion or malignancy . . . . .	24
5.2	Low-grade intraepithelial lesions . . . . .	25
5.3	High-grade intraepithelial lesions . . . . .	25
5.4	Squamous Cell Carcinoma . . . . .	25
6.1	Extracted “cells” . . . . .	30
6.2	Extracted “no cells” . . . . .	31
6.3	Cell recognition model architecture . . . . .	32
6.4	Low-grade Squamous Intraepithelial Lesion Example . . . . .	33
6.5	High-grade Squamous Intraepithelial Lesion Example . . . . .	33
6.6	Squamous cell carcinoma Examples . . . . .	34
6.7	Negative for Intraepithelial Lesion Examples . . . . .	34
6.8	Cell classification model architecture . . . . .	36
6.9	General Process . . . . .	39
7.1	Output . . . . .	40
7.2	Accuracy Results . . . . .	41
7.3	Precision Results . . . . .	42
7.4	Specificity Results . . . . .	42

7.5	Sensitivity Results . . . . .	43
7.6	140 predictions processing time (s) . . . . .	43

# Chapter 1

## Introduction

Of all the types of cancer, cervical cancer has one of the most significant death rates, being in the fourth position worldwide and mostly affecting less developed areas like Latin America and Africa. There are two methods to prevent cervical cancer, HPV vaccine and screening (Pap smear). Since most cervical cancer cases develop from HPV infections, providing HPV vaccines to women at young ages is crucial to reduce cervical cancer risk. Also, cervical cancer takes years to develop; that is why periodical screening is essential to detect this disease at early stages, incrementing possible treatments' effectiveness [1, 2]. Both mentioned methods have shown high efficacy, reducing mortality and incidence rates; for example, since the USA implemented widespread screening, cervical cancer cases have dropped more than 50% [3]. However, that is not the reality for all the countries; by 2008, the Institute for Health Metrics and Evaluation at the University of Washington showed countries with the highest global wealth decile achieved a 64% of women getting effective cervical screens, while countries with the lowest global wealth just reached a 9% [4]. Additionally, a projection made in 2019 shows that countries with low and medium human development indexes (HDI) will achieve just 10% and 20% of screening coverage by 2023; this could be the case of Ecuador, which was scored with a medium HDI of 0.759 by the United Nations in 2019 [5, 6]. In Ecuador, cervix cancer is the second with the most prominent incidence and mortality [7]. One of the obstacles in Latin America to improve the number of cervix cancer screens is the amount of time needed to give results. In Ecuador, laboratories take up to one month to give pap smear analysis results, and in Venezuela, this time can increase up to 3 months [8]. This problem with delayed results opened a new field where technologies can help to accelerate the process, especially Artificial Neural Networks (ANN) [9, 10]. In particular, neural network classification systems are optimal tools to speed up the screening process since cervical cancer screening is a matter of cell classification.

This thesis presents a fast cell recognition system with posterior classification using convolutional networks in TensorFlow ambient. Two main processes are involved in the proposed method: a low-resolution scanning for quick cell detection and isolation; and a high-resolution classification. The program categorizes cells into two types: Negative for intra-epithelial malignancy or Normal Cells and Abnormal Cells (which includes low and high squamous intraepithelial lesion and squamous cell carcinoma). The purpose of this classification is that the pathologist can save time by evaluating meticulously just those

samples with abnormal cells, incrementing the number of samples that can be processed. To achieve this, we parted from a database of 963 liquid-based pap smear images. These images corresponded to 460 patients and were taken with a 40x magnification using a Leica ICC50 HD microscope [11]. The challenge with this set of images is their classification. Samples are classified after the most abnormal cell found but do not imply that all cells in the sample fit in that category. Therefore, it is necessary to create a database by isolating individual cells and manually classifying them to train the neural networks. The present project develops an evaluation method that avoids robust categorization across all sample image sections and focuses this effort on essential areas to reduce processing time.

## 1.1 Objectives

### 1.1.1 General Objective

Develop a cell recognition system based on artificial intelligence that scans pap smear samples to identify abnormal cell presence using convolutional neural network prediction values.

### 1.1.2 Specific Objectives

- Find an accurate deep neural architecture that categorizes cells according to their medical importance.
- Find an efficient way to scan high-resolution images, minimizing the required computer processing.
- Create a system that recognizes abnormal cell presence from a liquid-based pap smear images dataset.

# Chapter 2

## Problem Statement

Cancer is a health complication in which cells grow without control. When this disease starts in the cervix, it is denominated cervical cancer [12]. Cancer is a leading cause of death worldwide, and cervical cancer is the fourth most common cancer type, causing almost 341,831 deaths in 2020 [13]. Most cervical cancer cases are related to human papillomavirus (HPV) infections. HPV infection is among the sexually transmitted with more incidence worldwide, But not all of these infections develop into cervical cancer. The Center for Disease Control and Prevention (CDC) specified that almost everyone gets HPV in the course of their lives, but 9 out of 10 disappear without intervention. However, HPV should not be underestimated since most cervical cancer cases are derived from HPV infections, especially from the high-risk subtype 16, which together with the 18 induce 70% of the cases [14, 15, 16].

The focal point to reduce cervical cancer incidence and death rates is how highly preventable it is. There are two prevention mechanisms: HPV vaccines and Pap test screening. Early cervical cancer stages present precancerous lesions; these lesions could last for years before developing cancer itself and, with correct treatment, are mostly curable. Therefore, screening is highly recommendable to detect cervical cancer at an early stage. Also, since most cervical cancer cases develop from an HPV infection, taking an HPV vaccine highly reduces the chances of developing cervical cancer [17, 18].

More than 200 HPV types have been discovered, and 13 present sufficient evidence to be classified as potential risks for cancer development; These are types 16, 18, 31, 33, 35, 39, 45, 51, 52, 56, 58, and 59. There exist three HPV vaccines, using recombinant DNA technology, to prevent contagion of high-risk HPV types. The 4-valent against types 6, 11, 16, and 18 available since 2006; the 2-valent against types 16 and 18 approved in 2007 and the 9-valent against types 6, 11, 16, 18, 31, 33, 45, 52, and 58 authorized in 2014. All three types are recommended between the ages of 11 to 26, being younger ages preferable. Also, all three types demonstrate the generation of immunity in 95% of cases, higher in men than in women by 6% [18, 19]. However, not everyone has access to an HPV vaccine; there is a clear socioeconomic differentiation. Countries with low HDI have notorious lower HPV vaccine coverage; in 2004, this sector hardly reached a 7% coverage, and there is a projection that by 2050 they will reach just 40%. Also, in Latin America, lower socioeconomic sectors do not have access to health services, and countries with the highest cervical cancer presence are the furthest behind the process [4, 20, 21].

There exist two types of screenings for cervical cancer prevention: Pap smear, also known as Papanicolaou test, and the HPV test. HPV test determines the presence of the HPV, while the Pap test checks for abnormal cells that may progress to cervical cancer. Professionals recommend that women start getting these kinds of screens at age 21; a doctor will determine its' frequency. The standard frequency for screens is 3 or 5 years for typical cases and more frequent if the patient's immune system is compromised [22, 23].

George Papanicolaou created pap smears in 1943. This type of screening requires collecting cell samples present in the cervical transition zone to determine if any of them presents characteristics of precancerous lesions [24]. The sample can be analyzed with two methods; liquid-based cytology, which conserves cells in a buffered alcohol solution; and conventional cytology, which transfers the collected cells directly to a slide, both methods being equally efficient. Pathologists and cytologists are the designated specialists to evaluate pap smear samples and determine if there is a morphological change in the collected cells; traditionally, the specialists manually evaluate each cell looking for abnormal features like nuclear enlargement, nuclear hyperchromasia, irregular nuclear contour, among others [23, 25]. If the pathologist finds some of the mentioned features, it classifies that cell in the following categories:

Low-grade Squamous Intraepithelial Lesion (LSIL): squamous cell presents changes that eventually may progress to a squamous cell carcinoma [23].

High-grade Squamous Intraepithelial Lesion (HSIL): squamous cell presents changes that represent a high probability of developing squamous cell carcinoma [23].

Squamous cell carcinoma: squamous cell represents a significant advance of cervical cancer [23].

Screens are an efficient resource to prevent cancer, but many people do not use them because of a bad socioeconomic or sociocultural situation; this is the case in Latin America. Pap smear screenings are available in Latin America, and governments have implemented many programs to promote them. However, cervical cancer incidence and mortality remain the same in most Latin American countries because pap smear screening programs do not reach the necessary population [18, 26, 27]. By 2008, low and middle-income countries, which comprise most of Latin America, reached just a 19% population coverage. Furthermore, a modeling study made in 2009 projects that those countries may not reach 90% screening coverage until 2050 [4, 28]. The present low coverage in pap smear screening is a consequence of the following three identified factors:

Sociocultural barriers: linking cervical cancer with promiscuity, men not allowing their wives to go under this procedure, lack of prevention culture, fatalistic attitude to the results preferring not to know them, and others [29].

Knowledge barriers: lack of knowledge about cervical cancer, the misconception of pap smear screening being painful, lack of knowledge about the purpose and benefits of pap smear screening, and others [29].

Health system barriers: lack of access to quality service, lack of medical supplies, mistreatment by the health staff, long waiting times, and others [29].

In this project, we will center on two health system barriers: waiting time and qualified personal. In Latin America, some patients decline to get pap screening because of the prolonged waiting time for the results delivery. By 2004, Public health services from Ecuador, Mexico, and Venezuela had reports of result turnaround times of up to 1, 2, and 3 months, respectively [8]. These prolonged waiting times translate into unconformity and

have remained through the years. In 2009, a survey made to 81 women from urban and rural areas of Ecuador showed that women were unsatisfied with the waiting time needed to receive their results, specifically when made in the public health system. Therefore, women with enough economic resources preferred to choose a private health system to shorten these times [30]. Later, a study performed in Guatemala, using data correspondent to 515 median age women from 2014, concluded that one reason for the patients' loss to follow-up their pap smear was the long turnaround times for the results [31]. Likewise, a study made in Peru in 2019 with 892 women between 18 and 49 years revealed that 32.4 % experienced a delay in the results delivery [32]. The Lancet Oncology Commission declared that improvement in long waiting times for diagnostic should be a top priority because they affect prevention and early initial treatments. These delays are usually related to lack of access to services, low quality of the histopathological test, and specialists burnout due to shortness of trained personal, especially in rural areas [33]. Traditionally, the professionals prepared to evaluate pap smear screens are clinical pathologists, anatomic pathologists, and cytologists. According to a report made in Ecuador in 2014, there were 386 specialists prepared to evaluate pap smear screens (referred to as Laboratory 2), representing just 1.62% of the medical staff, making a considerable contrast with other fields like general medicine, anesthesiology, and pediatry with 7278, 1682, and 1749 members [34]. A 2018 update of the previously mentioned report showed that Laboratory 2 personal increased to 408 members. However, in comparison to the whole medical staff, Laboratory 2 represented just 1.35%, showing a slight decrease [35] .

As mentioned, prolonged waiting times discourage women from getting their pap screens on time and obstruct an early diagnosis of cervical cancer, increasing its mortality. The lack of trained personnel to evaluate pap smear and the time it takes to evaluate each sample substantially worsen the problem. Technology development allies to reduce the mentioned problems and help improve the healthcare system. A clear reflection of that alliance is how artificial intelligence has fused with the medical field in the last years. Many neural networks and other systems have been presented to automate medical procedures, and much research has been done around them [36]. This project aims to be part of that technology-medicine alliance. Therefore, this thesis's principal contribution is implementing a two-stage cell classification focused on making a high-speed low-resolution scan to reduce the problem of prolonged waiting times for cervical pap tests and overdue diagnosis.

# Chapter 3

## Related Work

Song et al. presented a method to automate pap-smear cell evaluation using neural networks and image processing. The authors address the problem of overlapping cells, which can lead to misclassification of cervical cells. The document proposed a multi-scale convolutional network to enhance and draw missed cell contour pixels by recognizing cell features, especially the nucleus, and making predictions with nearby pixels. Overlapping cells are a considerable problem in pap-smear samples because they blur the cytoplasmic edge, whose deformation is a vital sign of intraepithelial lesions and carcinoma cells. Executing the proposed method with two different datasets demonstrated its efficiency. However, since the project has a robust implementation, the processing time per image was 12 minutes [37]. Then, this method has good accuracy but sacrifices processing time due to its high computation complexity.

Su et al. implemented a system for cervical cancer cell classification in samples of liquid-based cytology. The primary purpose of the presented method by the authors was to create a high-accuracy sort. To achieve the wanted accuracy, they used a cascade implementation with two different classification systems, one applied after another. Each system classifies evaluating distinct features like roundness, area, diameter, etc., all extracted with pixels counting and calculation rather than using neural networks, like other existent methods [38]. Although applying the cascade structure showed high accuracy because the classification systems complement each other, the proposed method focuses on manually isolated cells. Therefore, adding an extra technique to automatize cell extraction for later classification would be feasible.

Kumar et al. presented a framework for cancer detection and classification in 2015. The mentioned framework consisted of four processes: enhancing microscope images, segmenting background cells, extracting features, and classifying them. Kumar et al. used histogram equalization to modify image contrast in the image enhancement stage; K-means algorithm to perform segmentation; pixels calculations using their intensity, presence, and distribution to determine morphological features; and neural networks, random forest, among others for the classification stage [39]. This method differs from the previously mentioned because it works on the image as a whole instead of working in specific sections or elements.

Hussain et al. presented a cell classifier method based purely on nucleus analysis. The authors used convolutional networks and auto-encoder for nuclei segmentation and later

classification. The technique works on the whole cervical sample by predicting nuclei pixels and highlighting their contour to differentiate them when evaluating clustered nuclei. This project works directly in the given cervix sample, avoiding image preprocessing techniques, common in automated cell classification. According to their evaluation, the method's accuracy was 96% for categorization in normal and abnormal cells. However, it is worth mentioning that this method does not consider any other cell feature than the nuclei [40].

Shanthi et al. proposed a cervix cell classification system based on a neural network. The proposed model uses the pap smear Herlev databases and a convolutional neural network to extract cell image characteristics like edges, size, etc [41]. The authors used image processing techniques like Bi-histogram equalization, Sobel operator, brightness changes, etc., to highlight essential cell features like the nucleus and cytoplasm contour. The project uses supervised neural networks to classify cells into a minimum of two classes (normal and abnormal) and a maximum of five (normal, mild, moderate, and carcinoma), with high accuracies of above 92%. Experimental results were obtained by testing the neural networks in a group of isolated cell images [42].

# Chapter 4

## Technical Background

This section gives an overview of all the necessary concepts and relevant aspects of the project's development and artificial intelligence.

### 4.1 Neural Networks

Neural Networks (NN), also referred to as Artificial Neural Networks (ANN), are computational and mathematical attempts to simulate biological neural networks present in living creatures. However, the similarity between the source of inspiration and the human-made accomplishment now is reduced to how conceptually they work, mainly because no one understands how precisely biological neural networks function. Figure 4.1 shows a representation of a biological neuron. Biological neurons perceive electrical impulses and transmit them as signals across the neural body to the axon terminal, also known as the transmitter. Later, that impulse will reach the dendrites, or receptor, of a nearby neuron that will continue passing the information to the next neuron, forming a whole interconnected network able to make decisions, like brains. What artificial neural networks replicate is this concept of perceiving information that will later be transmitted across a network to make a decision [43].

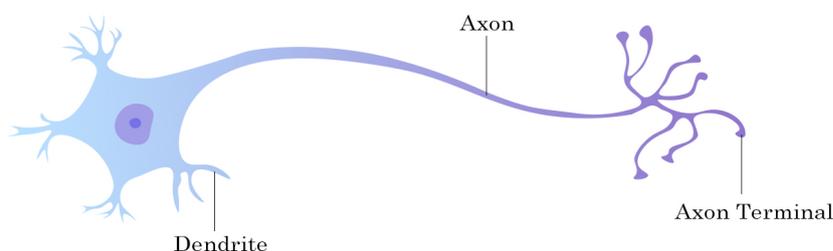


Figure 4.1: Biological Neuron.

Artificial neural networks use advanced linear algebra to replicate the biological model.

Also, for the same reason, they are structured as layers of neurons, connected to the following and previous layers. In addition, each layer has a particular neuron defined as a bias to collect a numeric value. The first layer is defined as the input layer since it receives all the data that later will change as it passes through the successive layers. The last layer is called the output layer since it returns the final result of all the modifications applied to the data through its journey in the network. Finally, all the remaining layers between the first and the last layer are called hidden layers; the number of hidden layers is usually directly proportional to the model's complexity. Figure 4.2 shows a representation of the mentioned structure [44].

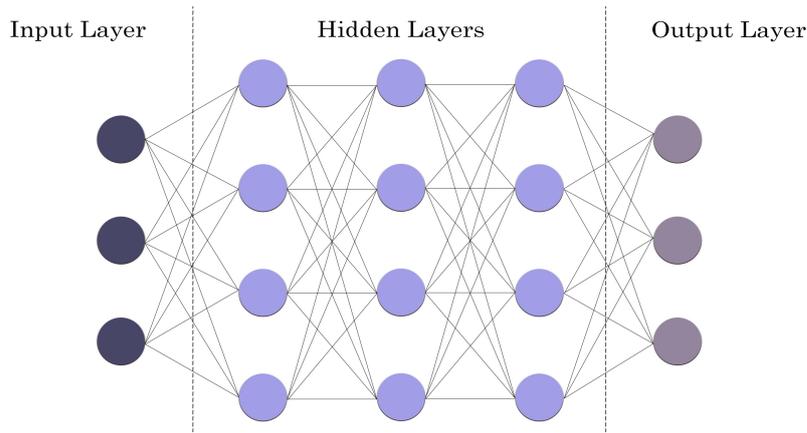


Figure 4.2: Neural Network Architecture.

All the data changes that occurred in the neural network are the result of Equation 4.1, called the weighted sum function; which is applied in each neuron [44].

$$S(w, b) = \left( \sum_{i=0}^n w_i x_i \right) + b \quad (4.1)$$

where:

$w_i$  is the weight of each neuron connection

$x$  is the value of the previous connected neuron

$b$  is the bias of each layer

$n$  is the number of connections

When applying only the weighted sum function, the neural network acts as a simple linear function or linear regression model. Therefore, it is necessary to apply the activation function to add complexity and dimensionality to the neural network. The activation function  $A(x)$  is applied to the whole weighted sum function as follows [45]:

$$A \left( \left( \sum_{i=0}^n w_i x_i \right) + b \right) \quad (4.2)$$

At the beginning of neural network execution, random biases and weights are defined, but as the neural network is training, it will gradually correct the values using the back-propagation technique. Finally, the results contained in the last layer are used to make predictions.

### 4.1.1 Hyperparameters

Hyperparameters are configuration values manually set before starting the neural network training and affect the learning process speed and final accuracy. Furthermore, the optimal hyperparameter values can not be calculated. Consequently, hyperparameters search is usually performed manually by try and error method until the model achieves a good performance [46].

#### Epochs

An epoch is passing the whole set of training data through the neural network once. Typically neural networks need several epochs to achieve good predictions because, in each epoch, weights and bias are modified to improve the model using backpropagation and gradient descent. This hyperparameter determines how many epochs will be executed to train the neural network [47].

#### Batch

Batch is a hyperparameter that determines how much data will pass through the network before the bias and weights are modified. The batch is classified into three types depending on its relation with the training dataset size [47].

- **Batch Gradient Descent:** Batch Size = Training Dataset Size
- **Stochastic Gradient Descent:** Batch Size = 1
- **Mini-Batch Gradient Descent:**  $1 < \text{Batch Size} < \text{Size of Training Set}$

### 4.1.2 Parameters

Parameters are configuration values estimated and modified during the neural network training.

#### Weight

Weights are used to represent the connection strength between two neurons of contiguous layers. All connections have weights involved and determine how much the neuron input will impact the output. The weight magnitude can be set before training or defined as a randomly generated value. Later, with the neural network training, weight values will change [48].

## Bias

Biases are numerical values that will also change when the neural network is training. This number's primary purpose is to shift the activation function allowing better predictions [43].

### 4.1.3 Activation Function

Activation functions usually are high degree functions applied to add complexity to the neural network. This function determines if a neuron can be fired or not. Also known as the transference function, their inclusion is vital to get complex predictions. There exist many options for activation function [49].

#### Binary Step Activation Function

This is the most basic activation function. It works as a threshold that activates the neuron if the input is higher than a defined value. Otherwise, the neuron is set to the inactive state, and the info is not passed to the next layer [45].

$$f(x) = \begin{cases} 0 & \text{if } x \leq \theta \\ 1 & \text{if } x > \theta \end{cases}$$

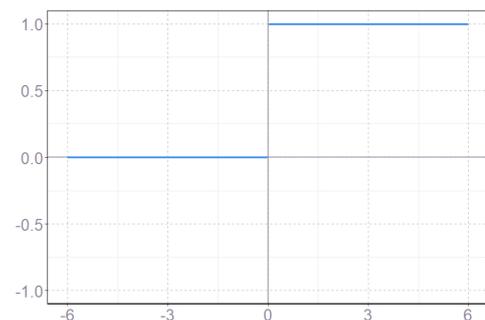


Figure 4.3: Binary Step Activation.

#### Linear Activation Function:

This function works by activating the neuron proportionally to the input. This function does not change much the neural network performance since the input value remains the same. Usually applied for simple tasks [45].

$$f(x) = x$$

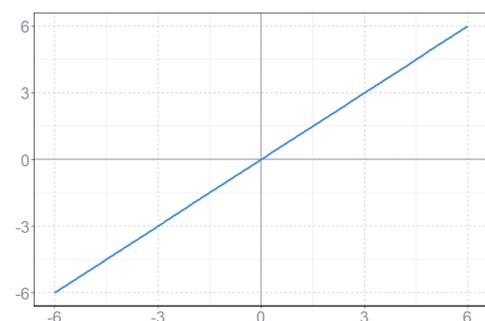


Figure 4.4: Linear Activation Function.

## Sigmoid Activation Function

This is the most used function and works by giving output values between zero and one. This activation function is not symmetrical about zero, and as a consequence, all output values will be positive. If necessary, this can be changed by scaling the function. It is typically used for binary classification [45].

$$f(x) = \frac{1}{1 + e^{-x}}$$

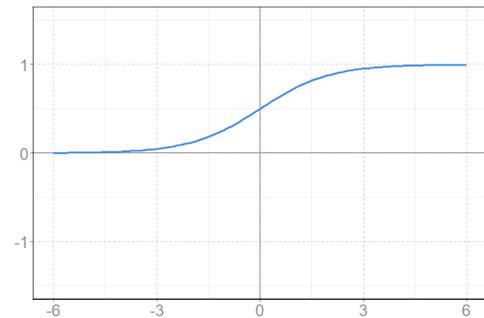


Figure 4.5: Sigmoid Activation Function.

## RELU Activation Function

Its name comes from “rectified linear unit.” This function is a non-linear activation function heavily used in neural networks because it has demonstrated better results than others. Its advantage against other activation functions is that it allows some neurons to be active while others are inactive at the same time. Contrary to other activation functions that activate all neurons [45].

$$f(x) = \begin{cases} 0 & \text{if } x \leq \theta \\ x & \text{if } x > \theta \end{cases}$$

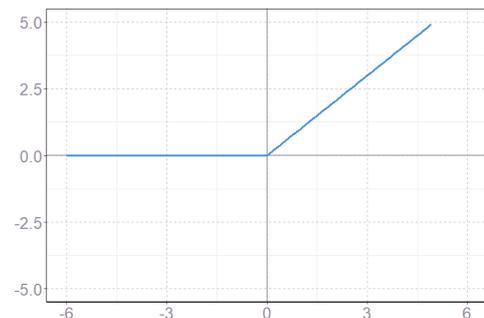


Figure 4.6: RELU Activation Function.

## Softmax Activation Function

This function is derived from the sigmoid function and has a similar common use. Since this function also returns output values between zero and one, they can be treated as probabilities for classification. Compared to the sigmoid function, the main difference is that the softmax function can be used for multiclass categorization. It is usually used in the last layer of multiclass classification neural networks to determine the percentage of prediction accuracy for each category [45].

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

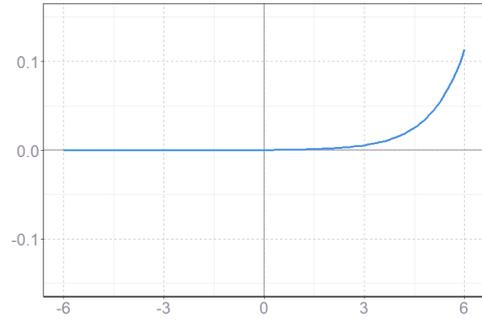


Figure 4.7: Softmax Activation Function.

Besides the ones presented, there are more viable activation functions, but most of them are adaptations of the already mentioned or are not commonly used [45].

#### 4.1.4 Loss Function

After information passes through the whole network and the activation function is applied, the last layer contains the information required to perform predictions. But, since the parameter values like bias and weights are randomly set at first, the first iterations will end up with terrible predictions. Predictions will gradually improve as the neural network gets trained. Then, it is necessary some way of controlling how good those predictions are. The Loss function is used to achieve this accuracy control [50].

As the neural network receives data, the loss function evaluates the difference between the final output and the received data's actual value. A high value in the loss function reflects terrible predictions and the need for more robust parameter modifications. Therefore, the objective while training neural networks is to reduce the loss function value as much as possible. For instance, many loss functions have been proposed to evaluate neural networks like mean squared loss, mean absolute loss, cross-entropy loss, etc. Cross-entropy loss is usually the preferred one for classification problems, while the mean squared loss is preferable for regressions [51, 52]. The Cross-entropy loss function has two versions; the equation 4.3 for models that make two class predictions and the equation 4.4 for models that make predictions for three or more classes. Mean squared loss is represented in equation 4.5.

Cross-entropy for  $m = 2$

$$CE(w, b) = -(y \log(p) + (1 - y) \log(1 - p)) \quad (4.3)$$

Cross-entropy for  $m > 2$

$$CE(w, b) = -\sum_{c=1}^m y_{o,c} \log(p_{o,c}) \quad (4.4)$$

where:

$m$  is the number of classes

$p$  is the probability of sample  $o$  belonging to class  $c$

$b$  is the binary indicator if the sample  $o$  belongs to class  $c$

$$MSE(w, b) = \left(\frac{1}{n}\right) \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (4.5)$$

where:

$\hat{y}$  is the value returned by the model

$y$  is the actual value

$n$  is the number of data point

### 4.1.5 Gradient Descent

Gradient descent is one of the keys for the automated learning present in neural networks. Gradient descent is the method used to automatically modify the neural network parameters, bias, and weights, to reduce the loss function result, improving neural network predictions. Reducing the loss function result can be translated to finding the minimum global point of the loss function. With a convex function, this minimum point can be located by setting the derivative of the variable's function to zero because the global minimum has a slope of zero. However, loss functions are usually non-convex and have multiple variable parameters, making the minimum point location a more complex task. Therefore, Gradient descent works by using each loss function parameter's partial derivatives to get their slope value. Then, those slope values are used as a reference to move towards lower values in the function as a matter of field reconnaissance, always looking for lower points. This process is carried several times until it reaches the possible minimum global point that represents more accurate neural network predictions [53, 54].

Each time the gradient descent is executed, bias and weights are modified to approximate the minimum loss function value. The learning rate defines the severity of changes in those parameters. Learning rate is usually a number between zero and one and is a crucial component to achieve the minimum global point of the function. If the chosen learning rate is too high, the gradient descent will act almost like a random number generator, and therefore the loss function result won't be reduced. On the contrary, if the learning rate is too low, the parameter changes will be so small that achieving the loss function reduction will take too much time and resources [53].

Assuming the chosen loss function for the model is the Cross-entropy function for  $m > 2$ , the gradients descent used to update weights, and bias value are represented as show in 4.6 and 4.7; where the derivatives are calculated with backpropagation.

$$\hat{w} = \hat{w} - \alpha \frac{\partial}{\partial \hat{w}} CE(w, b) \quad (4.6)$$

$$\hat{b} = \hat{b} - \alpha \frac{\partial}{\partial \hat{b}} CE(w, b) \quad (4.7)$$

where:

$\alpha$  is the learning rate

### 4.1.6 Backpropagation

Once we got the loss function results, we can determine how good the model predictions are. If the precision is terrible, then the model parameters must be changed. Backpropagation is a method used to determine how much each neuron is responsible for the process's final result. If the results are terrible, backpropagation helps identify which specific neurons are the principal accountable for that failure and then modify them with the correspondent method. In other words, backpropagation is the method used to calculate the partial derivatives of each parameter in the neural network with respect to the loss function; partial derivatives that are later used by the gradient descent to make the correspondent modifications of the parameter values [55]. The following three steps are the process that backpropagation executes repetitively from the last layer to the beginning of the network to obtain all the necessary derivatives.

- Calculate the error of the last layer.

$$\delta^L = \frac{\partial CE}{\partial A^L} \cdot \frac{\partial A^L}{\partial S^L} \quad (4.8)$$

- Backpropagate the calculated error to the previous layer.

$$\delta^{l-1} = w^l \delta^l \cdot \frac{\partial A^{l-1}}{\partial S^{l-1}} \quad (4.9)$$

- Calculate the derivatives of the current layer.

$$\frac{\partial CE}{\partial b^{l-1}} = \delta^{l-1} \quad (4.10)$$

$$\frac{\partial CE}{\partial w^{l-1}} = \delta^{l-1} A^{l-2} \quad (4.11)$$

### 4.1.7 Underfitting and Overfitting

Overfitting and underfitting are common problems that can appear when a neural network finished its training. Both cases lead to poor predictions. These problems are strongly related to the degree model, the training dataset, and the number of epochs used to train the model.

#### Underfitting

Some probable reasons a model is underfitting are very few epochs to train the model or using a model architecture too simple for complex problems. During each epoch, the model parameters are improved several times, depending on the batch size. If the model updates the original random values just a few times, the resulting model parameters would be closer

to the random values than the optimal values. Also, if the model's chosen architecture is too simple, it will reflect a low degree function for its predictions. Low degree functions have low variance, and high bias, which means the function results depend more on the bias than the training set [56]. Another possible cause of an underfitting model is using a lousy training dataset. If the data used to train the model does not represent their class's general characteristics, then the results accuracy is limited to samples similar to the used in the training set; failing in capturing the variability of the real data [57].

## Overfitting

On the other hand, overfitting could result from using unnecessary complex models or too many epochs. Using a complex architecture and, in consequence, a high degree function for predictions, the prediction results will depend too much on the training data, to the point in which the model doesn't acquire the relationship of the dataset but starts memorizing those specific samples [57]. This kind of model will probably have high accuracy when testing with the training data but much lower when testing with data never used in the training set. The same result will be obtained if the epochs used for training the neural network is too high [58].

## 4.2 Interconnection

According to how layers and neurons are connected in the neural network model, it can be classified into two major categories [59].

### 4.2.1 Feed Forward Connection

A neural network is classified as feedforward when each neuron of the layer fully connects to all the neurons of the following layer, having an assigned weight per neuron and a bias per layer [60]. Feed Forward connection is one of the most common models. In this kind of model, the input pass to the output deterministically, and depending on the number of layers, it can be multilayer or single-layer feed forward [61].

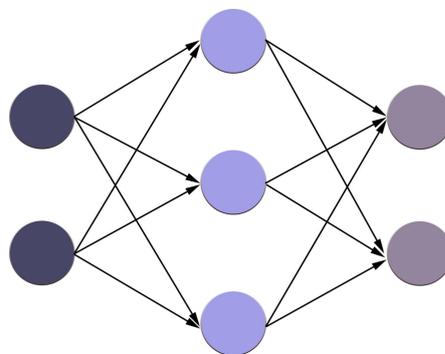


Figure 4.8: Feed Forward Connection.

## 4.2.2 Recurrent Network

Recurrent networks (RNN) are characterized because some neurons are connected to the following layer and a previous layer. Passing information again to an earlier neuron, or saving it for the subsequent sample analysis, gives this neural network architecture the capability of keeping some feedback of previous data. Therefore, recurrent networks do not make predictions of isolated samples, but predictions with the previous process's context [61]. RNN are good options for natural language and speech recognition, where the context of the last and following letters and sounds are essential to make a good prediction of the whole word [62].

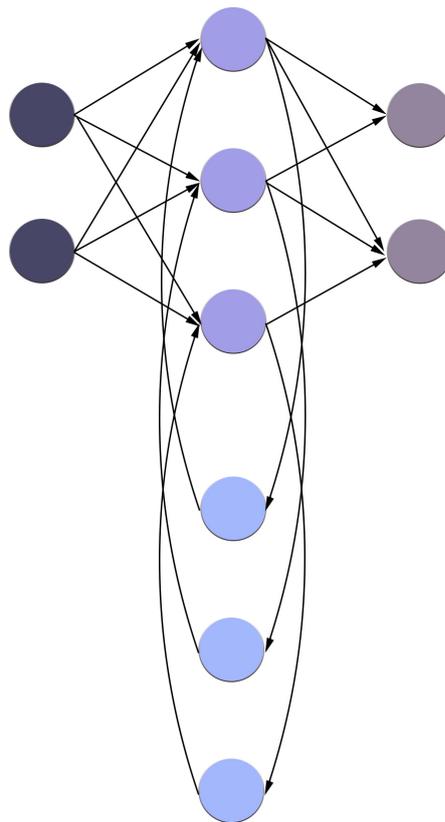


Figure 4.9: Recurrent Network.

## 4.3 Learning Rule

Based on the learning process and rules, neural networks are categorized as supervised, unsupervised, and reinforcement.

### 4.3.1 Supervised Learning

Supervised learning models create functions that use already classified data to train. Neural networks can be applied in very different areas. Then, for this type of model, a professional's

previous intervention in the applied field is necessary. A professional classifies all the training data according to their criteria, assigning labels. Later, the model is trained using the labeled data. Then, the trained model is applied to different data to verify it generalized the data characteristics properly [59, 63]. Supervised learning is a model primarily used, in many areas, for classification systems. In healthcare, for instance, a supervised neural network is used to identify human body lying postures and avoid pressure on ulcers [64].

### 4.3.2 Unsupervised Learning

Unsupervised learning models create functions that do not use classified data to train. In this case, the models focus more on recognizing hidden patterns or features in the evaluated data to create a way of organizing them [59]. A typical application of these models are recommendation systems, where customers are arranged in groups according to their purchases; to later use the obtained information for marketing [65].

### 4.3.3 Reinforcement learning

Reinforcement learning models create factions to take decisions based on the current situation of the problem. The neural network does not base its decisions on professionals' instructions or classifying information but on reaching for the best possible reward. During the training of this kind of learning method, the neural network exploits and explores its options. The model exploits the possibilities by preferring actions that produced great rewards in the past and explores by taking steps different from the past. After training, the model should use the information action-reward it recollected and generalized to decide which actions would create the most significant long-term reward [66]. This learning model is often used for games like chess, tic tac toe, or even modeling proteins [67, 68].

## 4.4 Convolutional Networks

Convolutional Neural Network (CNN) is a type of neural network commonly used for problems with high-dimensional data like images and videos. CNN works by applying filters of two or higher dimensions that perform convolution operations to the received input. CNN uses these filters because convolution operations detect features or structures in the evaluated image, which the CNN later uses to differentiate pictures from different categories [69]. The first approximation of a convolutional neural network was presented in 1980 by Kunihiko Fukushima, who inspired his work on the research made 30 years before by Wiesel and Hubel [70]. They studied animal cortexes' reactions to visual stimuli, discovering that brain neuron organization is based on layers. CNN is viable for supervised and unsupervised learning and usually has a feed-forward architecture. Since the CNN base is the classical structure of neural networks, most of the necessary concepts for its understanding are the same; however, the remaining topics related to CNN are treated in the following sections.

### 4.4.1 Filter

A filter, also referred to as kernel, is a matrix of numbers called weights; as seen in Figure 4.10. These weights work as used in traditional neural networks. CNN randomly initializes filter weights at the beginning of the training. Later, during the whole process, weight values are updated until they approximate the optimal conditions to make good predictions [71].

2	0
-1	3

Figure 4.10: Filter

### 4.4.2 Convolution Operation

Convolution is a mathematical operation that implies the multiplication of the filter and the input data, both being two-dimensional arrays [72]. These operations can detect structures in an image. During the pass of information in a convolutional neural network, neurons apply this operation in every output received from the previous layer. The original data received in the input layer is an image, which is an arrangement of bytes. Depending on the image, the array of bytes typically rank two or three; two for grayscale images and three for RGB images. During CNN execution, the array of bytes corresponding to the image is convolved with another array, called a filter. CNN applies this filter across the whole image, and the result is a new set of bytes, usually with smaller dimensions than the original image [73].

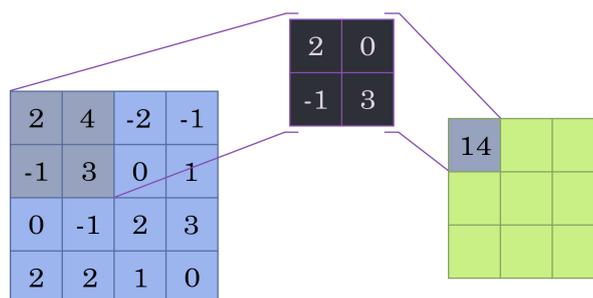


Figure 4.11: Convolution Example

### 4.4.3 Padding

The convolution operation is executed by the displacement of filters across the image, but this task encounters a border problem. When filters try to center pixels in the border,

the filter is out of range trying to convolute values outside the image. Therefore, either border values are ignored, or the Padding technique is implemented. Padding is a simple technique where all the area out of range is filled to allow convolution on the border. The most common padding version is zero-padding, where space is filled with zeros to allow convolution without excessively complicating calculations or deforming the original image [74].

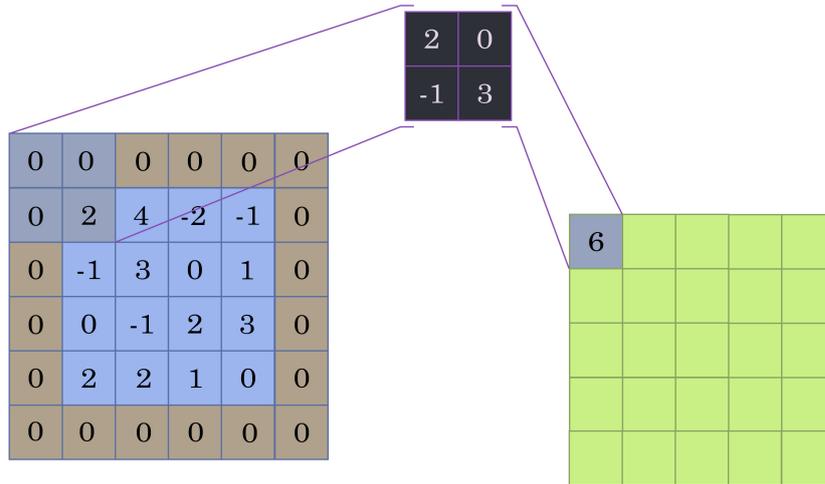


Figure 4.12: Padding Example

#### 4.4.4 Strides

Strides determine the distance a filter moves between its application; this distance or “steps” is measure in pixels. The most common stride value is one, which executes the filter on every step. However, other values can be set considering the more prominent the value, the more significant the information loss [75].

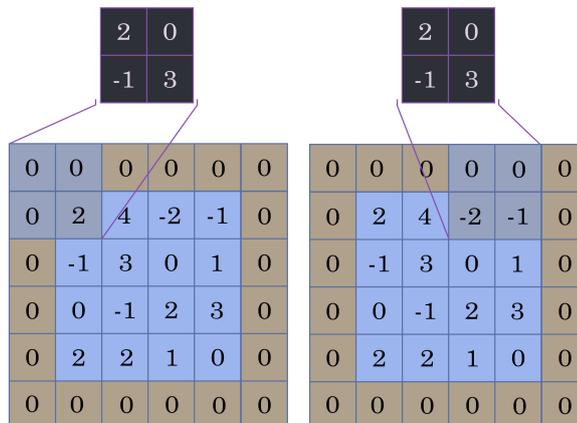


Figure 4.13: Stride of 3

### 4.4.5 Pooling

The pooling technique reduces the previous layer’s output to get a compact version, which conserves the essential information. Pooling, similar to convolution, uses filters of fixed dimensions to scan pixels across the whole image. The most common pooling techniques are Max Pooling, which takes the highest value, and Average Pooling that takes the pixels’ average. Furthermore, there exist other pooling techniques like Stochastic Pooling,  $L_p$  Pooling, Spectral Pooling, among others [76].

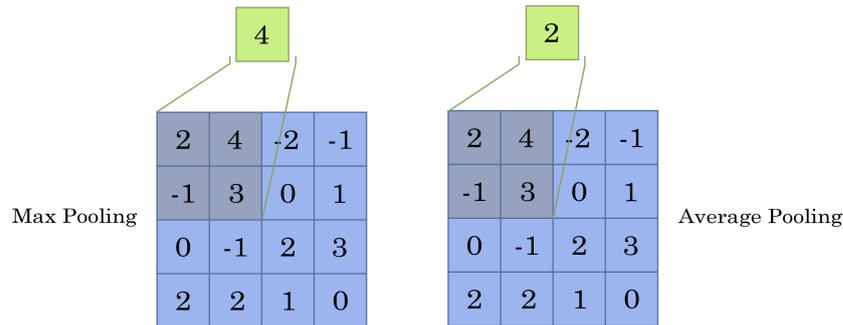


Figure 4.14: Max pooling vs Average Pooling

### 4.4.6 Dropout

As its name implies, Dropout refers to removing values in a neural network. The benefit of discarding some values while training a neural network is the overfitting risk reduction and implicitly exploring new architecture models. When a value or neuron is excluded, all its subsequent connections also disappear; therefore, it ends up “creating” a new architecture. The discarded neurons are chosen randomly by a fixed probability, so in each epoch, different neurons are deactivated. The dropout value should be chosen carefully because an excessive dropout incidence eliminates too much information, impeding the neural network from recognizing the image features [77].

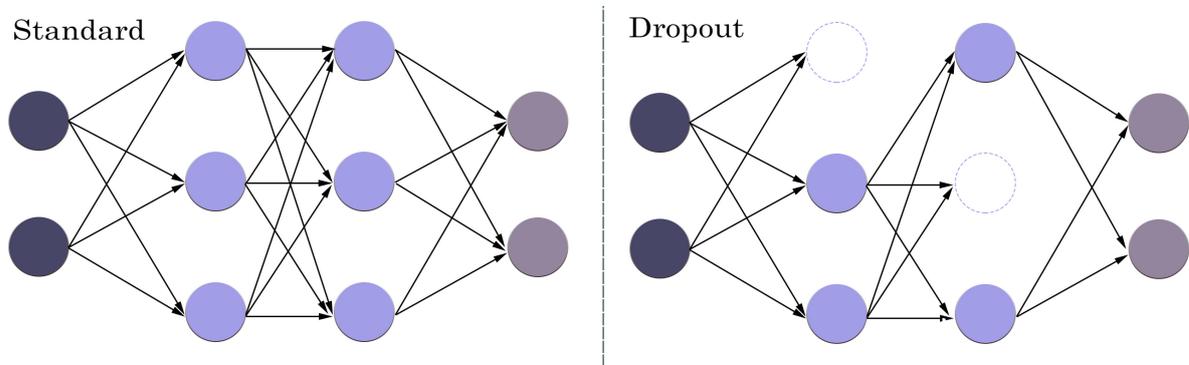


Figure 4.15: Standard vs Dropout

## 4.5 Data Augmentation

Data augmentation is an efficient technique to obtain more variability in the database. With this technique, the available images are duplicated and modified by flipping, zooming, rotating, cropping, among others; to create more data for the training process. Also, if the characteristics of the classification categories are well known, synthetic images can be generated by editing or even from scratch [78].

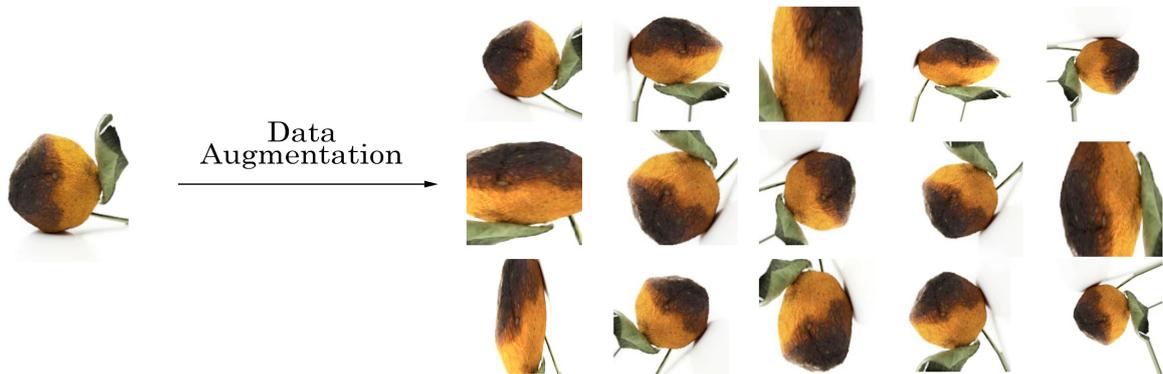


Figure 4.16: Data augmentation

## 4.6 Confusion Matrix

This matrix organizes the necessary information about predictions and actual categories to evaluate the classification system performance [79].

		Prediction	
		Positive	Negative
Actual Value	Positive	TP	FN
	Negative	FP	TN

Table 4.1: Cell Classification Model Results

As seen in 4.1 there are 4 possible options for the confusion matrix:

- True Positive (TP): Correct prediction about the evaluated element being in the positive class
- True Negative (TN): Correct prediction about the evaluated element being in the negative class
- False Positive (FP): Incorrect prediction about the evaluated element being in the positive class
- False Negative (FN): Incorrect prediction about the evaluated element being in the negative class

### 4.6.1 Confusion Metrics

#### Accuracy (AC)

Describes how many predictions were correct.

$$AC = \left( \frac{TP + TN}{TP + FP + FN + TN} \right) \quad (4.12)$$

#### Precision (P)

Describes how many of the elements labeled as positive are, in effect, positive.

$$P = \left( \frac{TP}{TP + FP} \right) \quad (4.13)$$

#### Sensitivity (SN)

Describes how many of the positive elements are labeled as positive.

$$P = \left( \frac{TP}{TP + FN} \right) \quad (4.14)$$

#### Specificity (SP)

Describes how many of the negative elements are labeled as negative

$$P = \left( \frac{TN}{TN + FP} \right) \quad (4.15)$$

# Chapter 5

## Data Description

The dataset contains 963 pap test images retrieved from the Mendeley data repository. All pap test samples were taken in 2019 from 460 patients at three Indian institutions:

- Babina Diagnostic Pvt. Ltd
- Dr. B. Borooah Cancer Research Institute
- Gauhati Medical College and Hospital

Samples were prepared using liquid-based cytology instead of the conventional method to obtain higher-quality images with cleaner backgrounds. Pictures were taken using a Leica DM 750 microscope connected to a specialized high definition camera ICC50 HD and its official computer software. Also, the microscope configuration was set to 400x magnification. All the mentioned conditions resulted in JPG high definition images with a size of 2048 x 1536 pixels [80].

An expert pathologist labeled each image into four categories, resulting in the following classification:

- Negative for Intraepithelial lesion or malignancy (NILM): 613 samples

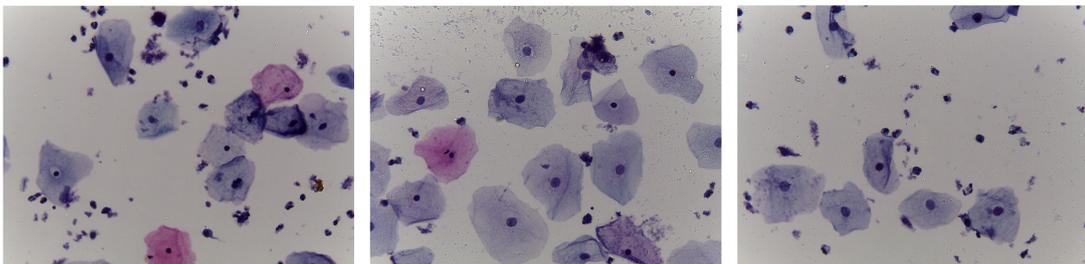


Figure 5.1: Negative for Intraepithelial lesion or malignancy

- Low-grade intraepithelial lesions (LSIL ) : 163 samples

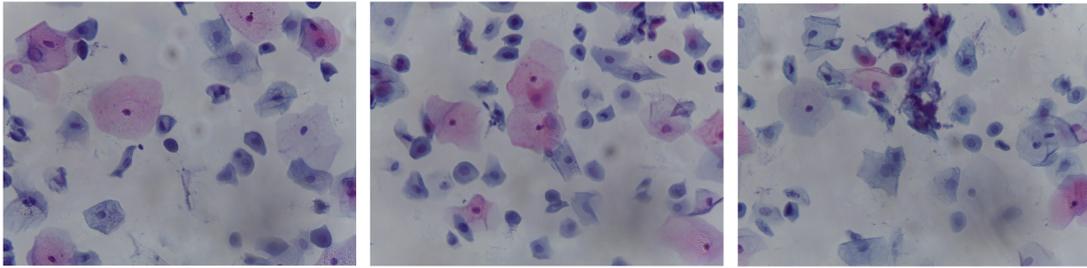


Figure 5.2: Low-grade intraepithelial lesions

- High-grade intraepithelial lesions (HSIL): 113 samples

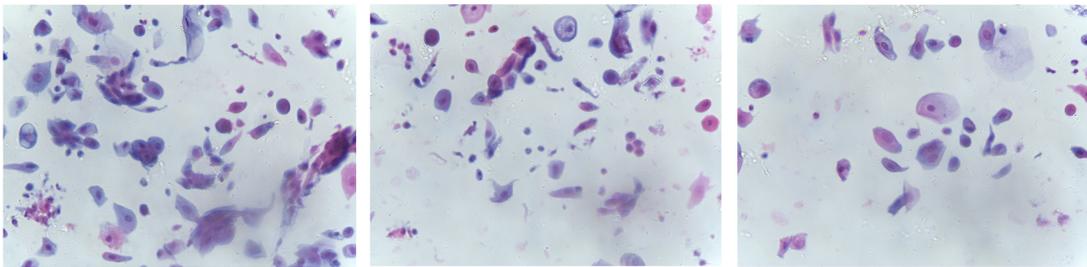


Figure 5.3: High-grade intraepithelial lesions

- Squamous Cell Carcinoma (SCC): 74 samples

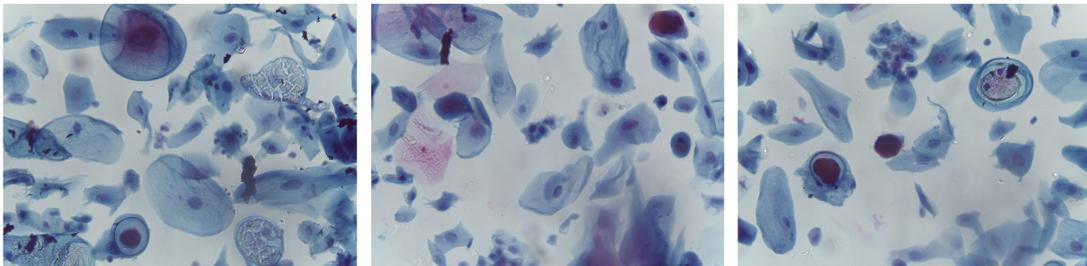


Figure 5.4: Squamous Cell Carcinoma

Pap test images contain an average of 20 cells, but it can increase up to approximately 50 cells. Thanks to the liquid-based technique employed, most pictures are “clean”, allowing good cell visibility.

# Chapter 6

## Implementation

This section describes the developed artificial intelligence system, which implements a fast scanning process for cervical cell recognition and posterior classification. The pap test images used for the project were retrieved from the Mendeley data repository and correspond to the liquid-based cytology of 460 patients. The original data-set image size is 2048 x 1536 pixels, but they were standardized to a width of 1080 pixels conserving the authentic width-height relationship, using the “Open cv-python” library. This reduction in the dimensions was performed to facilitate visualization since the used computer for coding had a resolution of 1920 x 1080 pixels, and original dimensions exceeded this limit. This modification was made to observe the process properly, but it is not essential in the proposed model.

A typical cervical pap test image contains an average of 20 cells, increasing up to approximately 50 cells. It is not efficient to train a convolutional neural network with images containing that amount of cells because they represent too many objects to evaluate at the same time and with too many details. The ideal case is to zoom in on individual cells, taking a closer look and assessing their characteristics to classify them in the available categories later, which is the process specialists perform during a pap smear examination. Therefore, this thesis’s main contribution is to develop a fast scanning process that takes a “quick look” with low resolution at the whole pap test sample and ubicate the coordinates of the possible target cells. That way, just the target cells are later zoomed in and fed to the high-resolution TensorFlow classifier model. The implemented low-resolution scanning allows individual cells to be zoomed in and isolated in a shorter time to focus the use of resources and time for a high-resolution CNN classifier.

### 6.1 Technology

#### 6.1.1 Software

The language used for the implementation is Python because it is versatile and with several implemented packages dedicated to scientific research [81]. By 2012, the most used languages in computational science were C/C++ and FORTRAN, with Python starting to gain recognition in the field [82]. However, the TIOBE index for April 2021 shows that

Python is the third most popular language, surpassed by Java in the second position and C in the first place. Also, the TIOBE index indicates that Python is one of the fastest-growing programming languages and has a percentage change to the date of +1.72, while C and Java percentage change values are -2.40 and -5.49. Besides, Fortran is in the twentieth position on the chart [83]. Python's principal disadvantage is its execution time, which is slower than C/C++. Nevertheless, Python is ideal for computational science programming since its characteristics allow coding time reduction, which is more beneficial for scientific prototyping [82]. Python also is used in research due to the variety of built-in libraries for data science, deep learning, image processing, cloud computing, machine learning, etc. Python even counts with its library version of OpenCV called "Open cv-python" developed by Intel and a specialized library for artificial intelligence called "TensorFlow" [84]. TensorFlow is used because it provides flexibility when setting parameters and hyperparameters in neural networks. Besides, TensorFlow is now one of the most used libraries for deep learning research due to all the tools and pre-build functions that facilitate neural network implementation [85]. Also, the Opencv library was used to display the analyzed images and track the process by making graphic representations. Even though Tensorflow has many pre-build tools to facilitate neural network implementation, those functions also difficult the data feed because they required a specific format to work. Furthermore, many of the TensorFlow functions change drastically from one version to another and, with them, the required format. Specifically, the versions used for this thesis were:

- python 3.8
- tensorflow 2.2.0 with
- opencv-contrib-python-4.4.0.46

### 6.1.2 Hardware

- Intel(R) Core(TM) i7-6500U CPU @ 2.50GHz, 2601 Mhz, 2 principal processors, 4 logic processors
- 12 GB RAM
- NVIDIA GeForce 940M

## 6.2 Parameters Setting

This section specifies all the parameters that need to be manually set for the fast scanning system, including those related to convolutional neural networks implementation, training, and the scanning process.

### 6.2.1 Region of Interest

The region of interest (ROI) refers to the current analyzed section of an image. Since the current thesis presents a fast cell scanning system, the ROI size must be set according to

the cell average size. For the developed model, the average cell size and the ROI size are 150 x 150 pixels. Certainly, this value concerns the 1080 width size pap test standardized measure, but the ROI size should be recalibrated if another dimension were used.

### 6.2.2 Batch size

The batch size defines how frequently weights and bias values in the convolutional neural network are updated. Both CNN models were implemented using a 32 mini-batch size since previous studies found that this and lower values yield better results [86]. Smaller batch size values were avoided because they exceeded the available hardware capacity, leading to problems during execution.

### 6.2.3 Epoch

Epoch is a hyperparameter that always depends on the whole neural network architecture. Therefore, there is no rule to choose the best value but try several ones until good results are obtained, avoiding overfitting and underfitting. In the case of the cell recognition model, with 100 epochs training, good predictions were obtained. While in the classifier model, 200 epochs were necessary to get decent predictions. It is worth mentioning that when training the classifier model, it showed overfitting from 373 epochs onward.

### 6.2.4 Activation Function

Activation functions depend on the task the neural network is performing. Based on a study made in 2017, the chosen function for all hidden layers in both models is the Relu activation function because it gives better results in most cases and is suitable for hidden layers [45]. The chosen activation function for the output layers was softmax because it is ideal for binary and multiple classifications, which meets the category classification requirements of both models.

### 6.2.5 Dropout

Dropout is a regularization technique essential for complex convolutional neural networks because it reduces or avoids overfitting probability. The dropout value determines the percentage of neurons that will be ignored; thus, values over 0.5 are avoided because they turn out sabotaging the model training [87]. Considering the mentioned information the classification model's dropout value was 0.3. On the other hand, the recognition model architecture does not include dropout layers to keep it as simple as possible to reduce its running time.

### 6.2.6 Maxpooling

Maxpooling is a technique used to lower overfitting by conserving just the neurons' group's most significant value. The value in the maxpooling filter determines the size of the evaluated group. Both models were set with maxpooling layers of 2 x 2 in both cases. Using

the current hardware without maxpooling layers in complex neural network architecture produced failures when running the algorithm due to the hardware limitation.

### 6.2.7 Filters

The dimensions of the images evaluated by the filters correspond to the standard size used to train each model. These models are explained in sections 6.3 and 6.4. Filters are a matter of try and error selection. However, using smaller filters usually detects small characteristics. Therefore the point of reference was to use small filters for the classifier model because cell features are essential to differentiate categories, choosing 3 x 3 filters for the 250 x 250 pixels images. On the contrary, the cell recognition model was set with more extensive filters because it does not need to identify all cell details to determine its presence, choosing 10 x 10 filters for the 20 x 20 pixels images. The number of filters was selected by trial and error, resulting in layers with 10, 32, 40, and 64 filters. The architectures' exact distributions are specified in 6.3 and 6.4.

### 6.2.8 Strides and Padding

Strides determine the steps filters move across the analyzed data, and padding completes the empty spaces with zeros when the filter stands in the image borders. For both cases, strides are set to 1 for vertical and horizontal displacement, which is the TensorFlow ambient's default value. Padding was deactivated since it's been shown that it affects the data variance and affects the CNN training [88].

## 6.3 Cell recognition Model Implementation

It is necessary for the fast cell recognition implementation to train a convolutional neural network capable of differentiating cells from the Pap test's empty areas. Therefore it is required to prepare the training and validation data and the convolutional neural network model.

### 6.3.1 Data Preparation

First, it is necessary to select several images from the liquid-based pap smear dataset. It is essential to choose pictures so that all the available categories are covered. Then, the photos are analyzed to recognize the average cell size and use that value as the ROI size in future processes; the average cell size selected was 150 by 150 pixels. Later, using the photo editor program of preference, sections of the images containing cells are manually cut out and saved. The photo editor program used for the current project was Adobe Photoshop 2020, and the criteria used to select the mentioned sections were:

- Sections that meet the ROI size.
- Areas containing a centered cervical cell, regardless of its condition.

- Areas containing cervical cell cluster.

Besides these images, sections containing centered but larger cells that exceed the ROI size limit are also selected. Also, a group of pictures was edited to have samples of completed isolated cells. All the extracted images are stored with the “cell” category label.

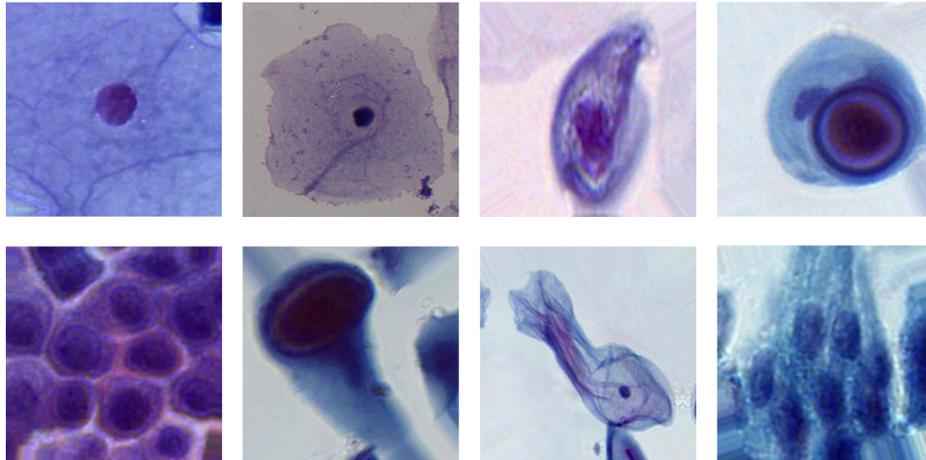


Figure 6.1: Extracted “cells”

A similar process is applied to extract images with the “no cell” category label. The difference eradicates in the used criteria, and that the ROI size restriction is maintained in all cases. Also, non-images from this category were edited. The criteria used were:

- Sections that meet the ROI size limit.
- Empty sections.
- Areas containing one or more not centered cervical cells.

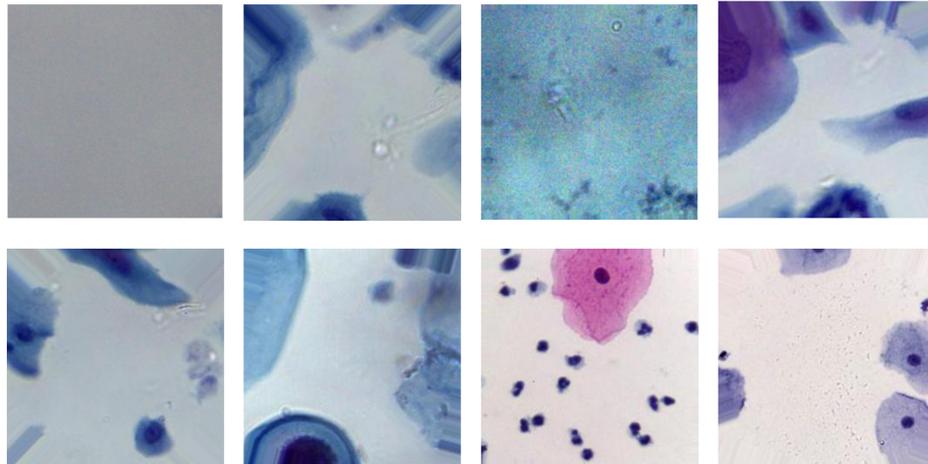


Figure 6.2: Extracted “no cells”

For the CNN to obtain good generalizations of the categories, it is necessary to choose enough images to represent as many cell shapes as possible. Then, data augmentation technique is used to generate more pictures and variability by zooming in, rotating, and flipping vertically and horizontally the images until it is enough to train the neural network. In this case, data augmentation was performed until it generated two thousand images for each category. Finally, 25 % of each type was saved for validation and the remaining for training.

### 6.3.2 Convolutional Neural Network Architecture

The convolutional neural network that recognizes cells needs to be as simple as possible to avoid using much hardware resources or time to make predictions [89]. Therefore, after some trials, the chosen architecture consisted of a neural network with four layers. The first one, a convolutional layer with ten filters of 10 x 10. The first layer is set to receive an input of 20 x 20 pixels. The second layer consists of a max-pooling layer of 2 x 2. The third layer is a flattened dense layer of 16 neurons. Finally, the output layer consists of 2 layers. The first and third layer uses a Relu activation function, while the output layer uses a softmax activation function. It is worth mentioning that even though using less complex models reduces training and testing time, implementing this technique in a compiled language like C can reduce execution time even more [90].

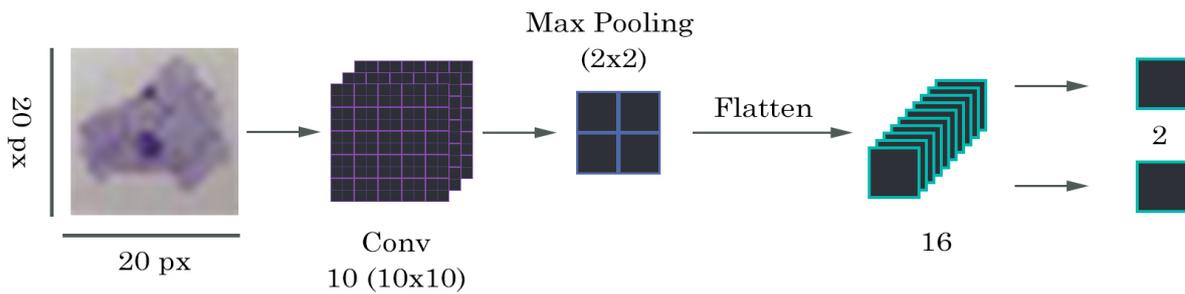


Figure 6.3: Cell recognition model architecture

### 6.3.3 Training

First, the training and validation images are rescaled to fit an interval of  $[0,1]$  and resized to 20 x 20 pixels. Then, images are transformed into tensors to fit the TensorFlow requirements and organized in batches of 32. Finally, training is performed during 100 generations, and the resultant model is saved. The CNN accuracy can be corroborated by testing it with the validation data by comparing predictions with the actual labels.

## 6.4 Cell classification model implementation

For cell classification, a convolutional neural network capable of categorizing cells in the following categories is implemented:

- Negative for Intraepithelial Lesion (NIL).
- Low-grade Squamous Intraepithelial Lesion (LSIL).
- High-grade Squamous Intraepithelial Lesion (HSIL).
- Squamous cell carcinoma (SCC).

Training data, validation data, and the convolutional neural network architecture are prepared first, similar to the cell recognition model.

### 6.4.1 Data Preparation

First, from the liquid-based pap smear dataset, random images are selected. It is crucial to choose pictures from the four categories, ensuring that enough cells' variability is included. Using the same ROI size as the cell recognition model, in this case, 150 x 150 pixels, cells are extracted, saved, and labeled in the correspondent category using the following criteria.

- Low-grade Squamous Intraepithelial Lesion (LSIL):
  - Slight nuclear enlargement
  - Irregular nuclear contour

- Hyperchromasia
- Slight cell shape variations

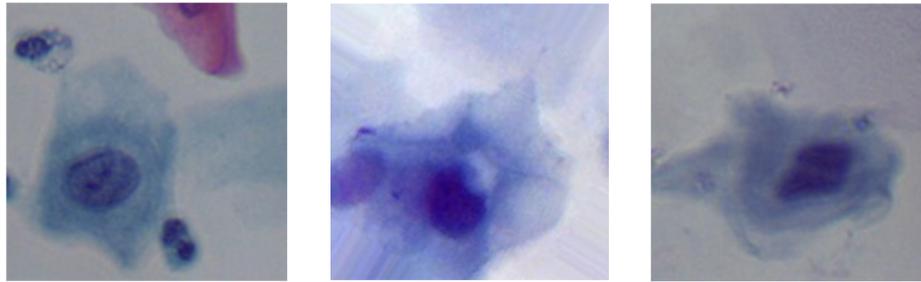


Figure 6.4: Low-grade Squamous Intraepithelial Lesion Example

- High-grade Squamous Intraepithelial Lesion (HSIL).
  - Marked Hyperchromasia
  - Hyperchromatic crowded group
  - Thick nuclear membrane
  - Marked irregular nuclear contour
  - Indented and irregular Nucleus
  - Usually parabasal-sized cell
  - Cell shape variations

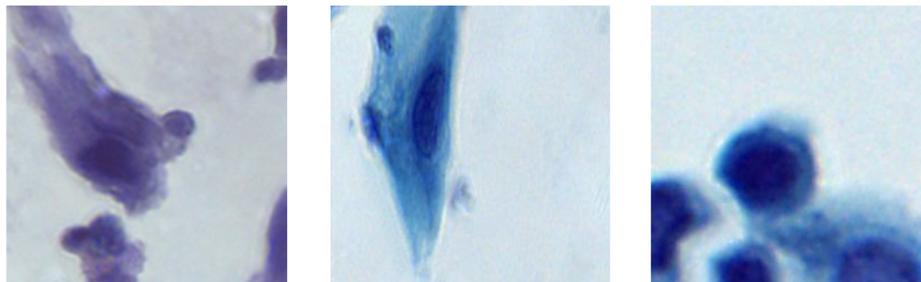


Figure 6.5: High-grade Squamous Intraepithelial Lesion Example

- Squamous cell carcinoma.
  - HSIL characteristics plus:
    - \* Macronucleus
    - \* Vacuolated cytoplasm
  - Tadpole cell
  - Fiber cell
  - Cell cluster with enlarged nuclei

- Marked cell shape variations



Figure 6.6: Squamous cell carcinoma Examples

- Negative for Intraepithelial Lesion (NIL):
  - Absence of the previously mentioned abnormal cell features

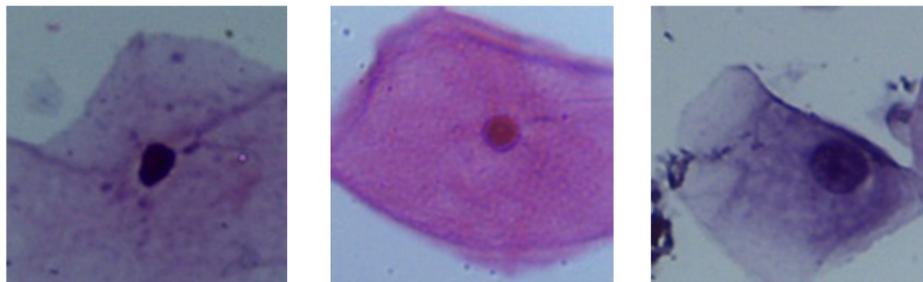


Figure 6.7: Negative for Intraepithelial Lesion Examples

Cells do not need to be centered to be selected. Also, in cases where the cell notoriously exceeds the ROI size limit, it can be ignored to capture the whole image, saving both the image where just part of the cell is appreciated and the one with the entire cell. In Addition, some pictures are edited to have samples of isolated cells. Due to the variability of the chosen images, these are later standardized to 250 x 250 pixels to avoid losing much information of bigger images without excessively incrementing information of images that meet the standard cell dimension.

The data augmentation technique is used to increase the number of samples, obtaining more variations of the chosen images. The mentioned technique applies zooming, rotation, and flipping to the dataset elements vertically and horizontally. In this case, data augmentation was performed until each category had 1200 images, except for the NIL, which had 2200 images. Finally, 200 images of each type were saved for validation and the remaining for training.

### 6.4.2 Convolutional Neural Network Architecture

The convolutional neural network to classify cells needs to be robust to recognize cell features and make better predictions. Therefore, the chosen architecture consists of more

layers and filters than the cell detection model. This convolutional neural network consists of 21 layers. The first one, a convolutional layer with 40 filters of 3 x 3. The first layer is set to receive an input of 250 x 250 pixels and three channels. The fourth and seventh are convolutional layers with 32 filters of 3 x 3 each one. The tenth, thirteenth and sixteenth are convolutional layers with 64 filters of 3 x 3 each one. The third, sixth, ninth, twelfth, fifteenth, and eighteenth are dropout layers set to eliminate 30% of the input. The second, fifth, eighth, eleventh, fourteenth, and seventeenth are max-pooling layers of 2 x 2. The nineteenth, twentieth, and twenty-first are a flattened layer of 128, 64, and 4 neurons. All the layers that require an activation function use a Relu activation function, while the output layer uses a softmax activation function.



### 6.4.3 Training

First, the training and validation images are rescaled to fit an interval of  $[0,1]$  and resize to 250 x 250 pixels. Like the recognition model, the images are transformed into tensors to fit TensorFlow requirements and organized in batches of 32. Finally, training is performed during 200 generations, and the resultant model is saved. The CNN accuracy can be corroborated by testing it with the validation data by comparing predictions with the actual labels.

## 6.5 Scanning Process

1. During the scanning process, several predictions are necessary; consequently, both previously trained CNN models are charged at the beginning of the process.
2. The Pap smear test file is loaded and resized to 1080 pixels to standardize all images but conserving the width-height relation allowing a correct visualization of the process on the monitor.
3. The whole matrix that makes up the picture is looped with steps of 70 pixels vertically and horizontally, extracting and storing matrix sections of ROI size (150 x 150 pixels). Also, the quadruple coordinates indicating where each of the pulled areas begins and ends are stored in a list B. Extracted images are stored in a list A to display them when desired. Nonetheless, displaying images is not essential for the procedure but helps track the process visually.
4. Two versions of each extracted section are stored as tensors to fit the CNN previously loaded TensorFlow requirements. Each version is stored in a different list to track them by their indexes; one version is resized to 250 x 250 pixels and stored in list C, and the other is resized to 20 x 20 pixels and stored in list D. Both are divided by 255.0 to rescale the tensor values to a  $[0-1]$  interval.
5. By looping every element of the list D, 20 x 20 pixels tensors are fed to the cell recognition CNN to start predictions. Whenever a forecast for "Cell" is thrown with an accuracy higher than 80%, list D's current index is used to track their corresponding image, coordinates, and 250 x 250 tensor version in lists A, B, and C; and store them in new lists to keep the index reference. Each time a cell is identified, coordinates in list B can be employed to draw a line in the Pap test sample image and have a visual representation of the process. The new lists for pictures, coordinates, and 250 x 250 pixels tensors are referred to as lists E, F, and G. This step should not consume much time since the predictions are made using a low-resolution CNN with few layers.
6. Now that all the 250 x 250 pixels tensor corresponding to possible cells are in list G, the list is looped similarly to the previous step using the classification cell CNN. Whenever a prediction corresponds to the following categories and accuracies, the coordinates on list F are used to draw their location with an indicative color.
  - Negative for Intraepithelial Lesion (NIL). accuracy > 70

- Low-grade Squamous Intraepithelial Lesion (LSIL). accuracy  $> 50$
  - High-grade Squamous Intraepithelial Lesion (HSIL). accuracy  $> 50$
  - Squamous cell carcinoma. accuracy  $> 80$
7. Finally, cells categorized as Negative for Intraepithelial Lesion (NIL) are labeled Normal Cells while the remaining categories are grouped in the Abnormal Cell Category.

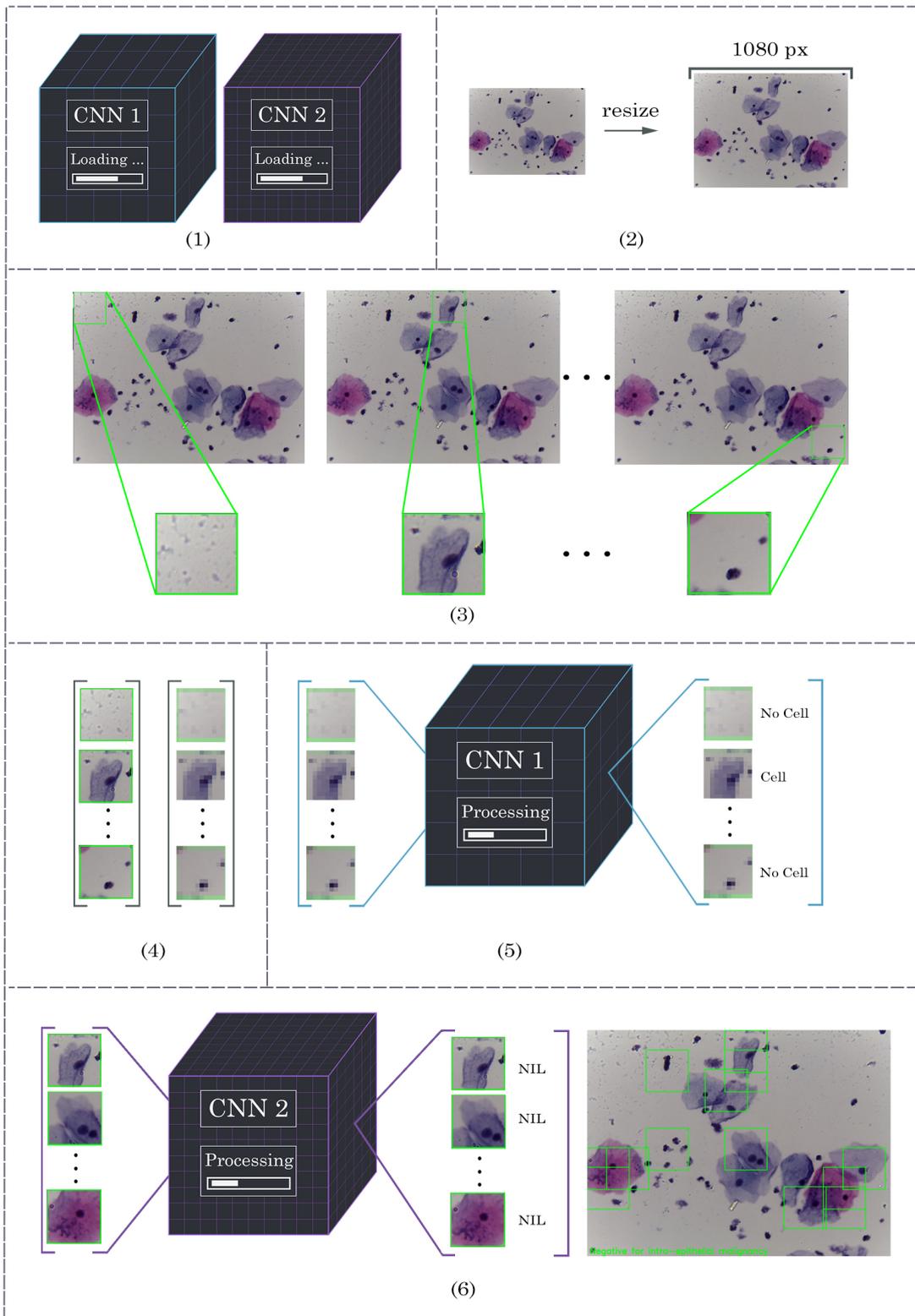


Figure 6.9: General Process

# Chapter 7

## Results

This chapter presents the results of the proposed two-stage cell classification system. Each time a pap smear sample is processed, the system saves an image with the respective classification, as seen in Figure 7.1. Therefore, the program was executed through the whole set of photos available in the Mendeley pap smear data-set, and all the output images were stored for analysis.

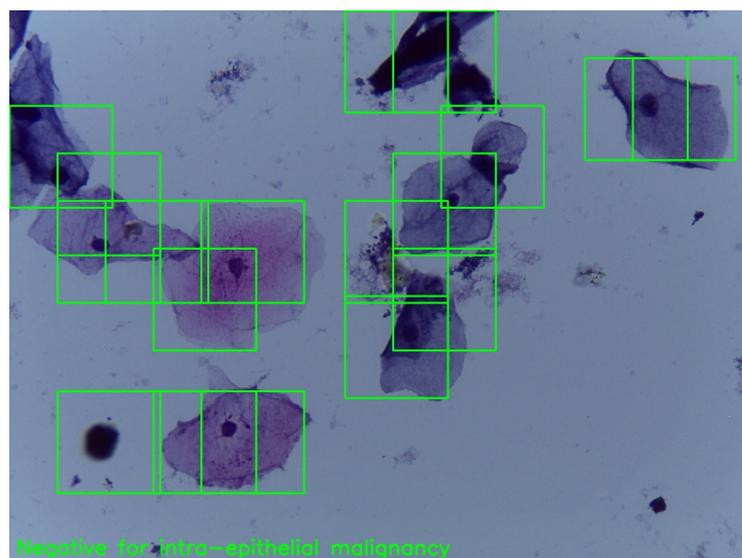


Figure 7.1: Output

This thesis focuses on implementing a system composed of a low-resolution high-speed convolutional neural network for cell recognition and a high-resolution convolutional neural network for classification, being the first one crucial for reducing pap smear results waiting times. Consequently, even though stages work together, the analysis was carried individually to spot their weaknesses and strengths.

The correspondent analysis was carried out using correct individual cell recognition and classification as successfulness indicators; instead of basing the examination on a generalized classification for the whole sample, as the original dataset does. Twenty random result images from each category were evaluated, and the obtained values were later fixed to fit

the actual dataset proportion, getting the results shown in Table 7.1 and Table 7.2. Then, these values were used to get the following indicators.

Cell Recognition			
True Positive	True Negative	False Positive	False Negative
7004	111848	716	2930

Table 7.1: Cell Recognition Model Results

Cell Classification			
True Positive	True Negative	False Positive	False Negative
479	5525	152	1218

Table 7.2: Cell Classification Model Results

Accuracy is an intuitive way of determining how well inputs were classified because it tells you how many images are correctly labeled from the whole group. As seen in 7.2 the cell recognition model reaches a value of 0.97, outperforming the classification model that got 0.81. This graph reflects the low-resolution recognition model correctly differentiated most empty spaces from actual cells or sections worth being analyzed; In contrast, the classifier model failed, labeling 20 percent of its inputs.

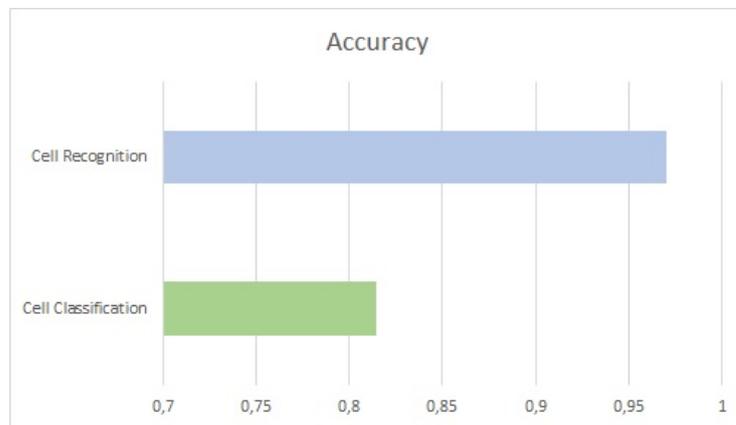


Figure 7.2: Accuracy Results

Precision value tells how many of the images categorized as cells by the recognition model and abnormal by the classification model actually belong to those categories. As appreciated in 7.3, the recognition model performs better than the classifier when predicting affirmative results. That value also means 10% of the images processed by the recognition model were later unnecessary analyzed by the categorizer because those images were not images of cells.

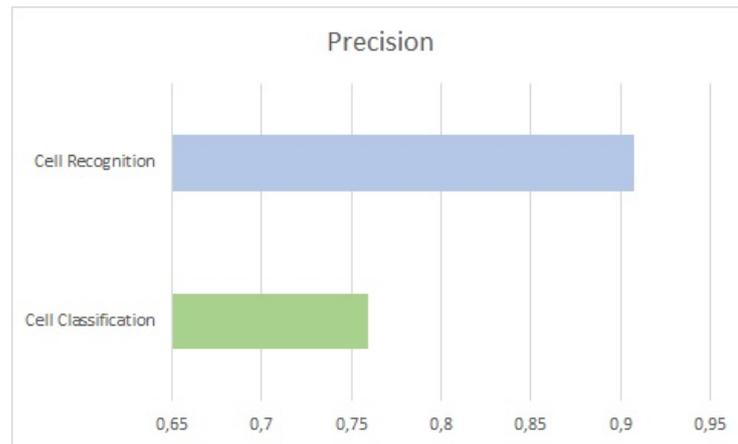


Figure 7.3: Precision Results

Specificity is an excellent indicator of how well our models work because its value reflects how many of the “no cell” images were categorized as such. In the same way, it indicates how many of the normal cells were classified correctly. As seen in 7.4 cell recognition model reached a specificity of 0.99 which means almost no empty region or irrelevant section from the pap smear was classified as a cell. Also, the classifier model specificity is 0.97 showing normal cells are rarely misclassified as abnormal.

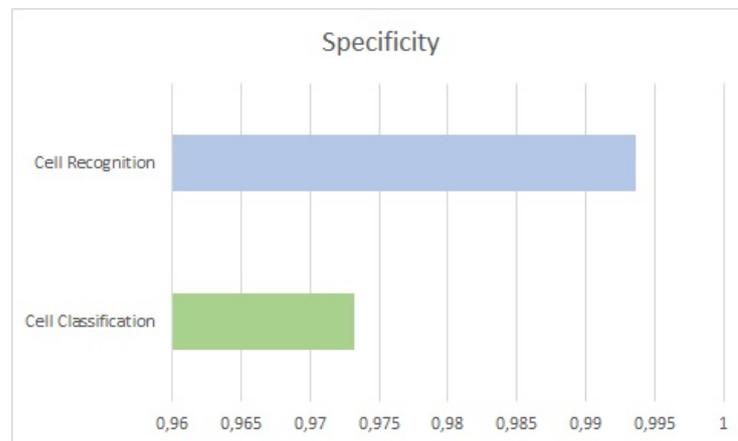


Figure 7.4: Specificity Results

Sensitivity is the complement indicator of specificity. In this case, sensitivity shows how many of the existing cells were recognized and how many of the abnormal cells were efficiently found. As seen in 7.5 there is a drastic difference between the two models. The low-resolution recognition model correctly found 70% of all the available cells while the classifier correctly labeled just 28% of the abnormal cells.

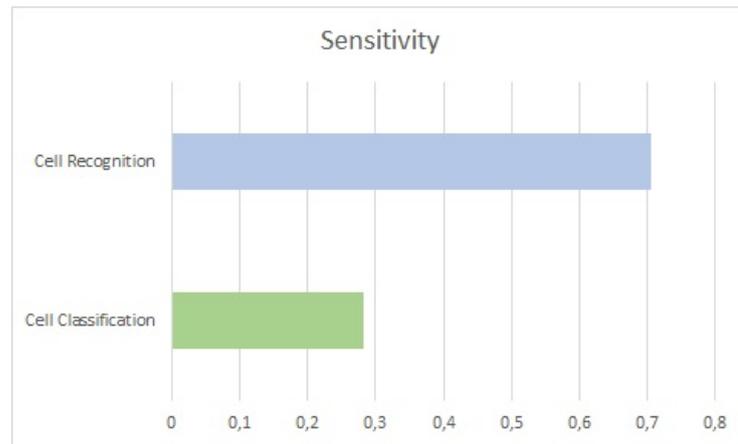


Figure 7.5: Sensitivity Results

Talking about time, 7.6 represents the amount of time it takes for both models to perform 140 predictions, which is the approximate value of necessary predictions per image. It is appreciated how the low-resolution recognition model needs less time to finish its predictions, and contrasting with the previous indicators information; it also performs better.

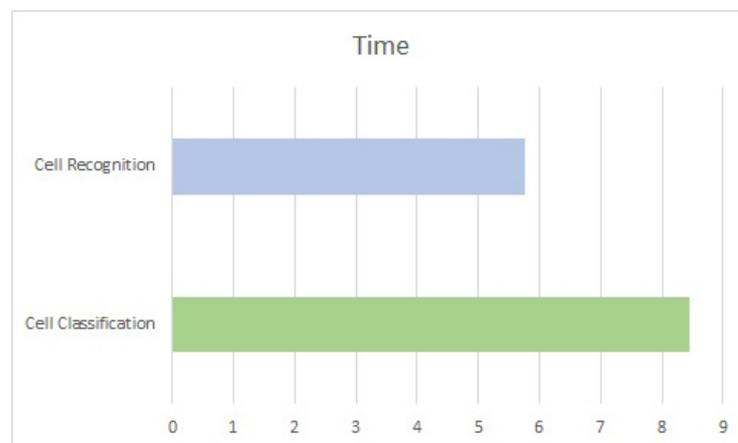


Figure 7.6: 140 predictions processing time (s)

# Chapter 8

## Conclusions

We conclude that it is essential to find a way to automatize pap smear scanning efficiently and reliably. Pap smear scanning continues to be a significant medical issue for millions of women, especially in developing countries where cervical cancer incidence is high, mainly due to the lack of prevention. Many women do not take their pap smear test due to unconformity with the process's long time.

Conventional pap smear scanning systems with high resolution consume too much computer resources and time, which difficult a quick diagnose of cervical cancer. Therefore, this thesis presents a solution to increase efficiency and reduce time consumption by using TensorFlow ambient to scan sample images with low resolutions faster. This method allows processing the same amount of images in less time.

The presented fast and low-resolution recognition system scans the sample image, makes fast predictions, extracts the sample sections containing cells or regions worth being checked meticulously, and passes them to a more robust classifier model. There, cells are categorized as normal or abnormal cells.

It is worth mentioning that the first stage of the system, which reduces the image resolution to make faster cell recognitions, achieved a 97% accuracy, 99% specificity, and 70% sensitivity, showing this is a viable and efficient way to reduce time consumption. On the contrary, the second stage that categorizes cells needs to be improved. It showed just 28% of sensitivity; this change can be achieved by a specialist's assistance when making the training database so that no cells are mislabeled, impeding a good training process of the CNN. Also, it is possible to increment the first stage's fastness with better hardware resources, especially with a better graphic card.

We conclude that the Tensorflow library is an excellent tool for research because it facilitates hyperparameters' configuration. Consequently, many neural network architectures can be implemented and tested in less time. However, the disadvantage we encounter using TensorFlow was that it is challenging to insert data into its ambient because the implemented functions are constantly changed, making some methods obsolete relatively quickly with their updates.

For future works, we recommend implementing image processing methods to highlight cell features like nuclei to make it easier for the neural network to detect those features. Also, we recommend improving the database to increase the classifier accuracy.

Finally, we conclude that the creation of automatic recognition systems is an excellent

option to meet the problem of long waiting times for pap smear results and, in consequence, to access early treatment, reducing cervical cancer incidence and mortality.

# Bibliography

- [1] World Health Organization. (2016) Screening as well as vaccination is essential in the fight against cervical cancer. [Online]. Available: <http://www.who.int/reproductivehealth/topics/cancers/fight-cervical-cancer/en/>.
- [2] K. Canfell, “Towards the global elimination of cervical cancer,” *Papillomavirus Research*, vol. 8, p. 100170, 2019.
- [3] The American College of Obstetricians and Gynecologists, “Practice bulletin no. 168: Cervical cancer screening and prevention, obstetrics & gynecology,” *Obstetrics & Gynecology*, vol. 128, pp. 111–130, 2016.
- [4] E. Gakidou, S. Nordhagen, and Z. Obermeyer, “Coverage of cervical cancer screening in 57 countries: Low average levels and large inequalities,” *PLoS Medicine*, vol. 5, 2008.
- [5] K. Simms, J. Steinberg, M. Caruana, M. Smith, J.-B. Lew, I. Soerjomataram, P. Castle, F. Bray, and K. Canfell, “Impact of scaled up human papillomavirus vaccination and cervical screening and the potential for global elimination of cervical cancer in 181 countries, 2020–99: a modelling study,” *PLoS Medicine*, vol. 20, 2019.
- [6] United Nations Development Programme. (2020) Human development reports. [Online]. Available: <http://hdr.undp.org/en/countries/profiles/ECU>
- [7] Global Cancer Observatory. (2020) Estimated age-standardized incidence and mortality rates (world) in 2020. [Online]. Available: <https://gco.iarc.fr/today/online-analysis-multi-bars>
- [8] I. Agurto, A. Bishop, G. Sánchez, Z. Betancourt, and S. Robles, “Perceived barriers and benefits to cervical cancer screening in latin america,” *Preventive Medicine*, vol. 39, no. 1, pp. 91–98, 2004.
- [9] N. Sompawong, J. Mopan, P. Pooprasert, W. Himakhun, K. Suwannarurk, J. Ngamvijcharoen, T. Vachiramon, and C. Tantibundhit, “Automated pap smear cervical cancer screening using deep learning,” in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 7044–7048.
- [10] J. Mangal, R. Monga, S. Mathur, A. Dinda, J. Joseph, S. Ahlawat, and K. Khare, “Unsupervised organization of cervical cells using bright-field and single-shot digital holographic microscopy,” *Journal of Biophotonics*, vol. 12, 2019.

- [11] E. Hussain, L. B. Mahanta, H. Borah, and C. R. Das, “Liquid based-cytology pap smear dataset for automated multi-class diagnosis of pre-cancerous and cervical cancer lesions,” *Data in Brief*, vol. 30, p. 105589, 2020.
- [12] Centers for Disease Control and Prevention. (2021) Basic information about cervical cancer — cdc. [Online]. Available: <https://www.cdc.gov/cancer/cervical/basic.info/index.htm>
- [13] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal, and F. Bray, “Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries,” *CA: A Cancer Journal for Clinicians*, p. caac.21660, 2 2021. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.3322/caac.21660>
- [14] P. A. Cohen, A. Jhingran, A. Oaknin, and L. Denny, “Cervical cancer,” *The Lancet*, vol. 393, pp. 169–182, 1 2019. [Online]. Available: <http://www.thelancet.com/article/S014067361832470X/fulltext><http://www.thelancet.com/article/S014067361832470X/abstract>[https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(18\)32470-X/abstract](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(18)32470-X/abstract)
- [15] E. J. Crosbie, M. H. Einstein, S. Franceschi, and H. C. Kitchener, “Human papillomavirus and cervical cancer,” *The Lancet*, vol. 382, pp. 889–899, 9 2013. [Online]. Available: <http://www.thelancet.com/article/S0140673613600227/fulltext><http://www.thelancet.com/article/S0140673613600227/abstract>[https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(13\)60022-7/abstract](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(13)60022-7/abstract)
- [16] Centers for Disease Control and Prevention. (2020) About hpv (human papillomavirus) — cdc. [Online]. Available: <https://www.cdc.gov/hpv/parents/about-hpv.html?>
- [17] J. Wang, C. X. Zheng, C. L. Ma, X. X. Zheng, X. Y. Lv, G. D. Lv, J. Tang, and G. H. Wu, “Raman spectroscopic study of cervical precancerous lesions and cervical cancer,” *Lasers in Medical Science*, pp. 1–10, 1 2021. [Online]. Available: <https://doi.org/10.1007/s10103-020-03218-5>
- [18] T. A. Kessler, “Cervical cancer: Prevention and early detection,” *Seminars in Oncology Nursing*, vol. 33, pp. 172–183, 5 2017. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/28343836/>
- [19] C. M. D. Oliveira, J. H. T. Fregnani, and L. L. Villa, “Hpv vaccine: Updates and highlights,” *Acta Cytologica*, vol. 63, pp. 159–168, 4 2019. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/30870844/>
- [20] K. E. Gallagher, D. S. LaMontagne, and D. Watson-Jones, “Status of hpv vaccine introduction and barriers to country uptake,” *Vaccine*, vol. 36, pp. 4761–4767, 8 2018.
- [21] S. Luciani, L. Bruni, I. Agurto, and C. Ruiz-Matus, “Hpv vaccine implementation and monitoring in latin america,” *Salud Publica de Mexico*, vol. 60, pp. 683–692, 11 2018. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/30699273/>

- [22] Centers for Disease Control and Prevention. (2021) What should i know about cervical cancer screening? — cdc. [Online]. Available: [https://www.cdc.gov/cancer/cervical/basic\\_info/screening.htm](https://www.cdc.gov/cancer/cervical/basic_info/screening.htm)
- [23] C. Mayer and D. P. Budh, “Abnormal papanicolaou smear,” 10 2020. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK560850/>
- [24] G. N. Papanicolaou and H. F. Traut, “Diagnosis of uterine cancer by the vaginal smear,” *New York*, vol. 46, 1943.
- [25] E. Cibas and B. Ducatman, *Citology: Diagnostic Principles and Clinical Correlates*. Saunders, 2014.
- [26] B. Nuche-Berenguer and D. Sakellariou, “Socioeconomic determinants of cancer screening utilisation in latin america: A systematic review,” *PLOS ONE*, vol. 14, p. e0225667, 11 2019. [Online]. Available: <https://dx.plos.org/10.1371/journal.pone.0225667>
- [27] M. S. Lopez, E. S. Baker, M. Maza, G. Fontes-Cintra, A. Lopez, J. M. Carvajal, F. Nozar, V. Fiol, and K. M. Schmeler, “Cervical cancer prevention and treatment in latin america,” *Journal of Surgical Oncology*, vol. 115, pp. 615–618, 4 2017. [Online]. Available: <http://doi.wiley.com/10.1002/jso.24544>
- [28] Organisation for Economic Co-operation and Development, “Dac list of oda recipients,” Tech. Rep., 2020. [Online]. Available: <http://www.oecd.org/dac/financing-sustainable-development/development-finance-standards/DAC-List-ODA-Recipients-for-reporting-2021-flows.pdf>
- [29] E. J. Liebermann, N. VanDevanter, M. J. Hammer, and M. R. Fu, “Social and cultural barriers to women’s participation in pap smear screening programs in low-and middle-income latin american and caribbean countries: An integrative review,” *Journal of Transcultural Nursing*, vol. 29, pp. 591–602, 11 2018. [Online]. Available: <http://journals.sagepub.com/doi/10.1177/1043659618755424>
- [30] Y. Godoy, C. Godoy, and J. Reyes, “Social representations of gynecologic cancer screening assessment a qualitative research on ecuadorian women,” *Revista da Escola de Enfermagem*, vol. 50, pp. 65–70, 2016. [Online]. Available: [www.ee.usp.br/reeusp](http://www.ee.usp.br/reeusp)
- [31] K. Austad, A. Chary, S. M. Xocop, S. Messmer, N. King, L. Carlson, and P. Rohloff, “Barriers to cervical cancer screening and the cervical cancer care continuum in rural guatemala: A mixed-method analysis,” *Journal of Global Oncology*, vol. 2018, 3 2018. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/35151515/>
- [32] A. F. Olaza-Maguiña and Y. M. D. L. Cruz-Ramirez, “Barriers to the non-acceptance of cervical cancer screenings (pap smear test) in women of childbearing age in a rural area of peru,” *ecancermedicalscience*, vol. 13, 2019. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/35151515/>

- [33] K. Strasser-Weippl, Y. Chavarri-Guerra, C. Villarreal-Garza, B. L. Bychkovsky, M. Debiasi, P. E. Liedke, E. S.-P. de Celis, D. Dizon, E. Cazap, G. de Lima Lopes, D. Touya, J. S. Nunes, J. Louis, C. Vail, A. Bukowski, P. Ramos-Elias, K. Unger-Saldaña, D. F. Brandao, M. E. Ferreyra, S. Luciani, A. Nogueira-Rodrigues, A. F. de Carvalho Calabrich, M. G. D. Carmen, J. A. Rauh-Hain, K. Schmeler, R. Sala, and P. E. Goss, “Progress and remaining challenges for cancer control in latin america and the caribbean,” *The Lancet Oncology*, vol. 16, pp. 1405–1438, 10 2015. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/26522157/>
- [34] Instituto Nacional de Estadística y Censos, “Anuario de estadística: Recursos y actividades de salud 2014,” Tech. Rep., 2014.
- [35] —, “Anuario de estadística: Recursos y actividades de salud 2018,” Tech. Rep., 2018.
- [36] E. R. Ranschaert, S. Morozov, and P. R. Algra, *Artificial intelligence in medical imaging: opportunities, applications and risks*. Springer, 2019.
- [37] Y. Song, E. L. Tan, X. Jiang, J. Z. Cheng, D. Ni, S. Chen, B. Lei, and T. Wang, “Accurate cervical cell segmentation from overlapping clumps in pap smear images,” *IEEE Transactions on Medical Imaging*, vol. 36, pp. 288–300, 1 2017. [Online]. Available: <https://ieeexplore.ieee.org/document/7562400/>
- [38] J. Su, X. Xu, Y. He, and J. Song, “Automatic detection of cervical cancer cells by a two-level cascade classification system,” *Analytical Cellular Pathology*, vol. 2016, 2016. [Online]. Available: [/pmc/articles/PMC4889791//pmc/articles/PMC4889791/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC4889791/](https://pubmed.ncbi.nlm.nih.gov/pmc/articles/PMC4889791/)
- [39] R. Kumar, R. Srivastava, and S. Srivastava, “Detection and classification of cancer from microscopic biopsy images using clinically significant and biologically interpretable features,” *Journal of Medical Engineering*, vol. 2015, pp. 1–14, 8 2015. [Online]. Available: [/pmc/articles/PMC4782618//pmc/articles/PMC4782618/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC4782618/](https://pubmed.ncbi.nlm.nih.gov/pmc/articles/PMC4782618/)
- [40] E. Hussain, L. B. Mahanta, C. R. Das, M. Choudhury, and M. Chowdhury, “A shape context fully convolutional neural network for segmentation and classification of cervical nuclei in pap smear images,” *Artificial Intelligence in Medicine*, vol. 107, p. 101897, 7 2020.
- [41] J. Jantzen, J. Norup, G. Dounias, and B. Bjerregaard, “Pap-smear benchmark data for pattern classification,” *Nature Inspired Smart Information Systems (NiSIS)*, 01 2005.
- [42] P. B. Shanthi, F. Faruqi, K. S. Hareesha, and R. Kudva, “Deep convolution neural network for malignancy detection and classification in microscopic uterine cervix cell images,” *Asian Pacific Journal of Cancer Prevention*, vol. 20, pp. 3447–3456, 2019.
- [43] K. L. Priddy and P. E. Keller, *Artificial neural networks: an introduction*. SPIE press, 2005, vol. 68.

- [44] G. Dreyfus, *Neural networks: methodology and applications*. Springer Science & Business Media, 2005.
- [45] S. Sharma, “Activation functions in neural networks,” *towards data science*, vol. 6, 2017.
- [46] M. Claesen and B. De Moor, “Hyperparameter search in machine learning,” *arXiv preprint arXiv:1502.02127*, 2015.
- [47] J. Brownlee, “What is the difference between a batch and an epoch in a neural network?” *Deep Learning; Machine Learning Mastery: Vermont, VIC, Australia*, 2018.
- [48] M. A. Kewalramani and R. Gupta, “Concrete compressive strength prediction using ultrasonic pulse velocity through artificial neural networks,” *Automation in Construction*, vol. 15, no. 3, pp. 374–379, 2006.
- [49] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation functions: Comparison of trends in practice and research for deep learning,” 12 2020.
- [50] F.-F. Li and A. Karpathy, “Compsci 682 neural networks: A modern introduction,” February 2013.
- [51] D. M. Kline and V. L. Berardi, “Revisiting squared-error and cross-entropy functions for training neural network classifiers,” *Neural Computing & Applications*, vol. 14, no. 4, pp. 310–318, 2005.
- [52] P. Christoffersen and K. Jacobs, “The importance of the loss function in option valuation,” *Journal of Financial Economics*, vol. 72, no. 2, pp. 291–318, 2004.
- [53] C. S. Vega, “¿qué es el descenso del gradiente? algoritmo de inteligencia artificial,” scenio, 2018. [Online]. Available: <https://www.youtube.com/watch?v=A6FiCDoz8.4&t=120s>
- [54] S. Ruder, “An overview of gradient descent optimization algorithms,” 2017.
- [55] Y. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, *Efficient BackProp*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 9–50. [Online]. Available: [https://doi.org/10.1007/3-540-49430-8\\_2](https://doi.org/10.1007/3-540-49430-8_2)
- [56] H. Jabbar and R. Z. Khan, “Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study),” *Computer Science, Communication and Instrumentation Devices*, pp. 163–172, 2015.
- [57] W. Koehrsen, “Overfitting vs. underfitting: A complete example,” *Towards Data Science*, 2018.
- [58] TensorFlow Developer Community, “Overfit and underfit,” 2020. [Online]. Available: [https://github.com/tensorflow/docs/blob/master/site/en/tutorials/keras/overfit\\_and\\_underfit.ipynb](https://github.com/tensorflow/docs/blob/master/site/en/tutorials/keras/overfit_and_underfit.ipynb)

- [59] R. Sathya and A. Abraham, "Comparison of supervised and unsupervised learning algorithms for pattern classification," *International Journal of Advanced Research in Artificial Intelligence*, vol. 2, no. 2, pp. 34–38, 2013.
- [60] L. O. Orimoloye, M.-C. Sung, T. Ma, and J. E. Johnson, "Comparing the effectiveness of deep feedforward neural networks and shallow architectures for predicting stock price indices," *Expert Systems with Applications*, vol. 139, p. 112828, 2020.
- [61] T. Jones, "Recurrent neural networks deep dive," 2017. [Online]. Available: <https://developer.ibm.com/articles/cc-cognitive-recurrent-neural-networks/>
- [62] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *CoRR*, vol. abs/1409.2329, 2014. [Online]. Available: <http://arxiv.org/abs/1409.2329>
- [63] T. Jones, "Models for machine learning," 2017. [Online]. Available: <https://developer.ibm.com/articles/cc-models-machine-learning/>
- [64] G. Matar, J.-M. Lina, and G. Kaddoum, "Artificial neural network for in-bed posture classification using bed-sheet pressure sensors," *IEEE journal of biomedical and health informatics*, vol. 24, no. 1, pp. 101–110, 2019.
- [65] O. Chang, G. Mosquera, Z. Castillo, and L. Zhinin-Vera, "Sales forecast by using deep rectifier network," in *Proceedings of the Future Technologies Conference*. Springer, 2020, pp. 378–389.
- [66] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [67] O. Chang, L. Zhinin-Vera, and F. Quinga-Socasi, "Self-taught neural agents in clever game playing," in *Proceedings of the Future Technologies Conference*. Springer, 2020, pp. 512–524.
- [68] O. Chang, F. A. Gonzales-Zubiate, L. Zhinin-Vera, R. Valencia-Ramos, I. Pineda, and A. Diaz-Barrios, "A protein folding robot driven by a self-taught agent," *BioSystems*, vol. 201, p. 104315, 2021.
- [69] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015.
- [70] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in *Competition and Cooperation in Neural Nets*, S.-i. Amari and M. A. Arbib, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1982, pp. 267–285.
- [71] J. Wu, "Introduction to convolutional neural networks," *National Key Lab for Novel Software Technology. Nanjing University. China*, vol. 5, p. 23, 2017.
- [72] J. Brownlee, *Deep learning for computer vision: image classification, object detection, and face recognition in python*. Machine Learning Mastery, 2019.

- [73] J. Cowley, “Convolutional neural networks,” 2018. [Online]. Available: <https://developer.ibm.com/technologies/artificial-intelligence/articles/cc-convolutional-neural-network-vision-recognition/>
- [74] M. Hashemi, “Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation,” *Journal of Big Data*, vol. 6, no. 1, pp. 1–13, 2019.
- [75] L. Zaniolo and O. Marques, “On the use of variable stride in convolutional neural networks,” *Multimedia Tools and Applications*, vol. 79, no. 19, pp. 13 581–13 598, 2020.
- [76] V. Passricha and R. K. Aggarwal, “Chapter 2 - end-to-end acoustic modeling using convolutional neural networks,” in *Intelligent Speech Signal Processing*, N. Dey, Ed. Academic Press, 2019, pp. 5–37. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780128181300000027>
- [77] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [78] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, “A guide to convolutional neural networks for computer vision,” *Synthesis Lectures on Computer Vision*, vol. 8, no. 1, pp. 1–207, 2018.
- [79] A. Santra and J. Christy, “Genetic algorithm and confusion matrix for document clustering,” *International Journal of Computer Science Issues*, vol. 9, 01 2012.
- [80] E. Hussain, L. B. Mahanta, H. Borah, and C. R. Das, “Liquid based-cytology pap smear dataset for automated multi-class diagnosis of pre-cancerous and cervical cancer lesions,” *Data in brief*, vol. 30, p. 105589, 2020.
- [81] K. J. Millman and M. Aivazis, “Python for scientists and engineers,” *Computing in Science Engineering*, vol. 13, no. 2, pp. 9–12, 2011.
- [82] M. G. Rashed and R. Ahsan, “Python in computational science: applications and possibilities,” *International Journal of Computer Applications*, vol. 46, no. 20, pp. 26–30, 2012.
- [83] TIOBE. (2021, 4) index tiobe the software quality company. [Online]. Available: <https://www.tiobe.com/tiobe-index/>
- [84] A. J. Dhruv, R. Patel, and N. Doshi, “Python: The most advanced programming language for computer science applications,” 2021.
- [85] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow:

- Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [86] D. Masters and C. Luschi, “Revisiting small batch training for deep neural networks,” *CoRR*, vol. abs/1804.07612, 2018. [Online]. Available: <http://arxiv.org/abs/1804.07612>
- [87] S. Park and N. Kwak, “Analysis on the dropout effect in convolutional neural networks,” in *Asian conference on computer vision*. Springer, 2016, pp. 189–204.
- [88] A. Nguyen, S. Choi, W. Kim, S. Ahn, J. Kim, and S. Lee, “Distribution padding in convolutional neural networks,” in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 4275–4279.
- [89] K. He and J. Sun, “Convolutional neural networks at constrained time cost,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5353–5360.
- [90] S. Nanz and C. A. Furia, “A comparative study of programming languages in rosetta code,” in *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, vol. 1, 2015, pp. 778–788.