UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Universitat Politècnica de Catalunya
Barcelona East School of Engineering
(EEBE)
Master's degree in Interdisciplinary and Innovative Engineering

DEEP LEARNING INTERPRETABILITY METHODS FOR THE CLASSIFICATION OF BLOOD CELL IMAGES

MASTER’S DEGREE FINAL PROJECT

Report and Annexes

Author: Steve A. Hernández Uptegrove
Director: Raúl Benítez Iglesias
Co-Director: Jose Julian Rodellar Benede
Convocatoria: September 2021

Abstract

During the past decade, the Medical Sector has widely adopted Neural Networks as a tool to help diagnose and to further understand different diseases. This is due to their proven high accuracy and versatility. However, its integration into the pathologists' workflow has been severely affected due to the black box nature these models present. The complex mathematical and statistical concepts these models are based on greatly hinder the direct understanding of the model's decision criteria when these perform predictions. Neural Network Interpretability aims to provide explanations in understandable terms to a human.

In this project, a deep learning interpretability study is carried out on DisplasiaNet, a Convolutional Neural Network specially optimized to classify Peripheral Blood Neutrophil images into Dysplastic or Normal. Working closely with expert pathologists and with the help of a purposely built web annotation app, the main morphological characteristics of the different cell states are extracted. Image interpretability techniques such as Saliency Maps, Class Activation Maps, and Occlusion Sensitivity Maps are applied to DisplasiaNet to obtain the features the model considers the most relevant.

The study has found that DisplasiaNet detects dysplasia in Neutrophils in a similar manner to expert pathologists, thus validating its accuracy. Firstly it focuses on the granularity of the cytoplasm, and secondly on the nucleus chromatinic density and lobular segmentation.

Resum

Durant l'última dècada, el sector mèdic ha adoptat les xarxes neuronals com a eina per ajudar a diagnosticar i comprendre diferents malalties, degut a la seva elevada precisió i versatilitat. No obstant, la seva integració al flux de treball dels patòlegs s'ha vist greument afectada per la naturalesa "Black-Box" que presenten aquests models. Els complexos conceptes matemàtics i estadístics en què es basen aquests models, dificulten enormement la comprensió directa dels criteris de decisió en el qual es basen per fer les seves prediccions. La interpretabilitat de xarxes neuronal té com a objectiu proporcionar explicacions en termes comprensibles a un ésser humà.

En aquest projecte, es duu a terme un estudi d'interpretabilitat a la xarxa DisplasiaNet, una xarxa neuronal convolucional especialment optimitzada per classificar les imatges de neutròfils sanguinis perifèrics en Normals o Displàstics. Treballant estretament amb patòlegs i amb l'ajut d'una aplicació d'anotacions web construïda a propòsit, s'extreuen les principals característiques morfològiques dels diferents estats cel·lulars. En paral·lel, s'apliquen tècniques d'interpretabilitat d'imatges a la xarxa DisplasiNet, com ara mapes de saliència, mapes d'activació de classes i mapes de sensibilitat envers l'oclusió, per obtenir les característiques que el model considera més rellevants.

L'estudi ha descobert que DisplasiaNet detecta displàsia en neutròfils de manera similar als patòlegs, validant així la seva precisió. En primer lloc, es centra en la granularitat del citoplasma i, en segon lloc, en la densitat cromatínica del nucli i la segmentació lobular.

Resumen

Durante la última década, el sector médico ha adoptado ampliamente las redes neuronales como una herramienta para ayudar a diagnosticar y comprender diferentes enfermedades. Sin embargo, su integración en el flujo de trabajo de los patólogos se ha visto gravemente afectado debido a la naturaleza "Black-Box" que presentan estos modelos. Los complejos conceptos matemáticos y estadísticos en los que se basan estos modelos dificultan enormemente la comprensión directa de los criterios decisivos que el modelo emplea para realizar predicciones. La interpretabilidad de redes neuronales tiene como objetivo proporcionar explicaciones en términos comprensibles para un ser humano.

En este proyecto, se lleva a cabo un estudio de interpretabilidad de la red neuronal DisplasiaNet, una red convolucional especialmente optimizada para clasificar imágenes de neutrófilos de sangre periférica en displásicas o normales. Trabajando en estrecha colaboración con patólogos expertos y con la ayuda de una aplicación de anotación web expresamente diseñada, se extraen las principales características morfológicas que presentan los diferentes estados celulares. En paralelo se aplican a DisplasiaNet técnicas de Interpretabilidad de redes neuronales especializadas en el análisis de imágenes tales como Mapas de relevancia, Mapas de activación de clases y Mapas de sensibilidad de oclusión para obtener las características que el modelo considera más relevantes.

El estudio ha encontrado que DisplasiaNet detecta displasia en neutrófilos de manera similar a los patólogos expertos, validando así su precisión. En primer lugar, se centra en la granularidad del citoplasma y, en segundo lugar, en la densidad cromática del núcleo y la segmentación lobular.

Acknowledgments

First and foremost, I would like to thank my tutors Raul Benitez and Jose Rodellar, for all the time spent throughout the numerous meetings to help me structure and guide me with this project.

I cannot express my gratitude enough to Andrea Acevedo for allowing me to continue working on DisplasiaNet and helping me gain profound knowledge on the motivations, procedures, and inner workings of the model.

A sincere thanks also goes out to Dr. Anna Merino from the Hospital Clinic de Barcelona. The insightful view given from the perspective of a Medical Doctor has allowed me to further understand the required symbiosis between engineering and science in order to solve real-life medical problems. The many explanations given to me regarding different Haematological aspects have awakened a fascination for biological interactions. This reality has also motivated me to pursue and obtain a more profound knowledge on this matter and learn how to apply Machine Learning to aid in diagnosing diseases faster and more accurately.

Special gratitude goes out to my classmates, especially Valentino Asole, for the many hours spent debating, problem-solving, and keeping each other company while grinding away on the project.

With this project, I finalize my Master's degree in Interdisciplinary and Innovative Engineering. A profound, heartfelt thanks goes out to my family: Amadeu, Sue, Christie, Brian, Elena and, Baldo, to whom without their unconditional support, I would not have been able to complete.

Table of contents

ABSTRACT ____ I

RESUM ____ II

RESUMEN ____ III

ACKNOWLEDGMENTS ____ IV

TABLE OF CONTENTS ____ V

GLOSSARY ____ VIII

LIST OF FIGURES ____ IX

LIST OF TABLES ____ XI

LIST OF EQUATIONS ____ XII

1. INTRODUCTION ____ 13
   - 1.1. Background 14
   - 1.2. Project objectives 15
   - 1.3. Project Scope 15
   - 1.4. Understanding Dysplasia in Neutrophils 16
     - 1.4.1. What are the Myelodysplastic syndromes? 16
     - 1.4.2. The Neutrophil 18
     - 1.4.3. Morphology of a Normal Neutrophil 18
     - 1.4.4. Morphology of a Dysplastic Neutrophils 19
   - 1.5. Understanding DisplasiaNet 21
     - 1.6.1. The database 21
     - 1.6.2. Architecture 23
     - 1.6.3. Evaluation and Result Interpretation 26
   - 1.7. Understanding Neural Network Interpretability 28
     - 1.7.1. Why is Neural Network interpretability needed? 28

2. STATE OF THE ART ____ 32
   - 2.1. Interpretability techniques 32
     - 2.1.1. Feature Analysis 33
     - 2.1.2. Model Inspection 33
     - 2.1.3. Saliency Maps 34
     - 2.1.4. Occlusion Sensitivity Maps 35
     - 2.1.5. Proxy Methods 36
   - 2.2. NN interpretability in Medical Applications 36

3. METHODOLOGY ____ 38

3.1. Interpretability study dataset ... 38
3.2. Cell annotation APP ... 39
3.3. Feature Maps ... 41
3.4. Low dimensionality projection of the deep features ... 43
3.5. Saliency Maps ... 44
3.6. Occlusion techniques ... 45
3.6.1. Occlusion Sensitivity Maps (OSM) ... 46
3.6.2. Local Interpretable model-agnostic explanations(LIME) ... 48
3.7. Class Activation Maps (CAM) ... 48
3.7.1. Grad-CAM ... 49
3.7.2. Grad-CAM ++ ... 50
3.7.3. Score-CAM ... 50
3.8. Background generalization test ... 52
4. SOFTWARE IMPLEMENTATION ... 53
4.1. Cell Annotation App ... 53
4.1.1. GUI Application Description ... 53
4.1.2. Annotation procedure ... 55
4.1.3. Docker Installation ... 57
4.2. Interpretability Study ... 58
5. RESULTS ... 61
5.1. Cell Annotation App results ... 61
5.2. DisplasiaNet Score vs Cell Annotation APP score ... 62
5.3. Low dimensionality projection of the Deep features ... 64
5.4. Feature Maps ... 65
5.5. Saliency maps ... 66
5.6. Occlusion Sensitivity Techniques ... 68
5.6.1. OSM: Occlusion Sensitivity Map ... 68
5.6.2. LIME: Local interpretable model-agnostic explanations ... 70
5.7. Class Activation Map (CAM) Techniques ... 72
5.7.1. Grad-CAM ... 72
5.7.2. Score-CAM ... 73
5.8. Background Generalisation Test ... 74
6. DISCUSSION ... 76
7. CONCLUSION AND FUTURE WORK ... 78
8. REFERENCES ... 80
ANNEX A. CELL ANNOTATION APPLICATION INSTRUCTIONS ... 84
ANNEX B. ANNOTATION DATA + DISPLASIANET PREDICTION ... 88

ANNEX C. IMAGE COLLECTION 89
C1. Study Dataset Images 90
C2. Saliency Vanilla Maps 92
C3. Smoothgrad Saliency Maps 94
C4. Occlusion sensitivity Maps 96
C5. LIME 102
C6. Grad-CAM Maps 103
C7. Grad-CAM++ Maps 106
C8. Score-CAM Maps 108

ANNEX D. RESULTS OF THE BACKGROUND GENERALIZATION TEST 110

Glossary

AI- Artificial Intelligennce

AML - Acute Myeloblastic Leukemia

ANN- Artificial Neural Network

API- Application Programming Interface

APP- Application

BM – Bone Marrow

CAM- Class Activation Mapping

CNN – Convolutional Neural Network

ConvNet – Convolutional Networks

EDTA - Ethylene Diaminetetraacetic acid

FLOPs – Floating Point OPerations

GDPR- General Data Protection Regulation

GUI – Graphical User Interface

LIME- Local Interpretable model-agnostic explanations

MDS - Myelodysplastic Syndrome

MGG - May Gründwald-Giemsa

ML – Machine Learning

NN – Neural Network

OSM – Occlusion Sensitivity Maps

PB – Peripheral blood

ReLU – Rectified Linear Unit

RGB – Red Green Blue

ROC - Receiver Operating Characteristic

VPN – Virtual Private Network

WHO – World Health Organization

List of Figures

Figure 1. Hematopoietic cell classification source: [9] ······ 17

Figure 2.A. Healthy Neutrophil. (TN_SNE_14872673.jpg) B. Healthy Neutrophil. (SNE_131893). Source: [14] ······ 19

Figure 3.A. Agranular Cytoplasm (SNE_1901668.jpg) B. Dohle body (MMY_780452.jpg) C. Dohle Body (SNE_769084.jpg). Source: [14] ······ 19

Figure 4.A.PseudoPelger-Huët Nucleus.(WD_SNE_2683958.jpg) B. Hiposegmented Nucleus (WD_SNE_2426196.jpg). Source: [14] ······ 20

Figure 5. A.Hipersegmented Nucleus (SNE_1896407.jpg) B. Ring Nucleus (SNE_741882.jpg) C. Band Nucleus (BNE_1841329). Source: [14] ······ 20

Figure 6. Examples of the dataset images: dysplastic cell(right), normal cell (left). The Neutrophil cell is in the center of the image. Red blood cells conform the majority of the background. Platelets can occasionally be observed, represented as smaller lilac colored isolated structures. Source: [14] ······ 22

Figure 7. DisplasiaNet Architecture. Source:[3] ······ 24

Figure 8. ROC curve analysis and truth table for the Proof of concept dataset. Source: [3] ······ 27

Figure 9. Cell Annotation APP GUI ······ 39

Figure 10. Feature Maps of a Normal Cell (TN_SNE_118039.jpg) ······ 42

Figure 11. Comparison Between Vanilla saliency maps (top) and Smoothgrad Saliency Maps (Bottom) ······ 45

Figure 12. Occlusion Sensitivity Maps patch iteration ······ 46

Figure 13. OSM grey patch 5px, 10px, 20px, 40px [TD_BNE_2249981.jpg (Top), TD_BNE_2256328.jpg (Bottom)] ······ 47

Figure 14. Quickshift superpixel generation (TD_BNE_2249981.jpg) ······ 48

Figure 15. Grad-CAM methodology scheme (Source: [23]) ······ 50

Figure 16. Score-CAM methodology scheme (source:[36]) ······ 51

Figure 17.Two Neutrophils in the same Image, with and without background (ZN_SNE_4223913.jpg) ······ 52

Figure 18. Cell Annotation APP GUI ······ 54

Figure 19. Cell Annotation APP GUI - Display Card ······ 54

Figure 20. Cell Annotation APP GUI - Tools Card ······ 55

Figure 21. User Annotation Workflow ······ 56

Figure 22. Docker desktop GUI. 1- Name of the Docker Container and Status, 2- Open Container on browser, 3- Open CLI, 4- Stop container, 5- Restart Container, 6- Delete Container from computer ······ 58

Figure 23. Images relabeled by the Expert pathologist. Left:Relabeled Normal(WD_SNE_2288170.jpg) Center: Relabeled Normal(WD_SNE_2426212.jpg) Right: relabeled dysplastic(WN_BNE_3328133.jpg) ······ 62

Figure 24. Graph comparing the score from the annotation applications VS the Score obtained from DisplasiaNet. Total Score(left), Cytoplasm Score(Center), Nucleus Score(right) ······ 63

Figure 25. t-SNE map combined with the 4 class images and overlaid with the nominal score of the image given by the expert pathologist. ······64

Figure 26. Difference between a Dysplastic cell and a Normal Cell - Polling layer 3- channel 5. Left (TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg) Right (TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg) ······66

Figure 27. SmoothGrad Saliency maps for Dysplastic Neutrophils (top) and Normal Neutrophils(Bottom)······67

Figure 28. Dysplastic Nuclear Aberrations ······67

Figure 29. OSM average patch 5px, 10px, 20px, 40px [TD_BNE_2249981.jpg (Top) TD_BNE_2256328.jpg (Bottom)] ······68

Figure 30. OSM average patch 5px, 10px, 20px, 40px [ZN_SNE_4198926.jpg (Top) ZD_BNE_5456886.jpg (Bottom)] ······69

Figure 31. Lime Results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg) ······71

Figure 32. Grad-CAM results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg) ······72

Figure 33. Grad-CAM++ results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg) ······73

Figure 34. Score-CAM results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg) ······74

Figure 35. Dysplastic Neutrophil (ZD_SNE_2759718.jpg) source: [6] ······75

List of Tables

Table 1. Dataset for training and testing DysplasiaNet CNN. source[3] ········ *26*

Table 2. Interpretability image dataset. ········ *39*

Table 3. Summary of Scores extracted with Cell Annotation APP ········ *62*

Table 4. OSM summary table ········ *70*

Table 5. Summary of the background generalization test ········ *75*

List of Equations

(Eq. 1) Rectified Linear Unit (ReLU) activation function 25

(Eq. 2) Sigmoid activation function 26

(Eq. 3) Gradient backpropagation for NN 44

(Eq. 4) Smoothgrad Saliency Map general equation 45

(Eq. 5) Noise vector definition for Smoothgrad Saliency Map 45

(Eq. 6) Relation between noise level and pixel distribution for Smoothgrad Saliency Map 45

(Eq. 7) Class Score gradient definition 49

(Eq. 8) Feature map weight for Grad-CAM 49

(Eq. 9) ReLU Activation applied onto the weighted feature maps for Grad-CAM 49

(Eq. 10) Feature map weight for Grad-CAM++ 50

(Eq. 11) Feature map normalization for Score-CAM 51

(Eq. 12) Masked input images to be processed with Score-CAM 51

(Eq. 13) Feature map weight for Score-CAM 51

(Eq. 14) ReLU Activation applied onto the weighted feature maps for Score-CAM 51

1. Introduction

With the continuous rise of machine learning techniques and advancements in hardware, more and more models are being deployed into production environments. Leveraging data to make significant decisions will directly affect individual lives in many sectors such as healthcare, justice, finance, education, marketing, and even human resources.

Regarding the healthcare sector, artificial intelligence (AI) has trickled a whole new dimension of diagnostic and prevention tools to aid specialists in treating and monitoring patients. AI developers must set special attention to ensure a safe, ethical and responsible practice of such technology. Looking ahead, AI has the potential to drive bigger changes and to having an extensive impact on society.

While pathologists can use this technology to assess and predict medical outcomes, their integration has been critically limited due to the difficult task of interpreting the black-box nature of these models. In a sense, a neural network (NN) is trained by providing a generic model with both the question and the answer. Through the iterative process of forward and backward propagation, the weights and biases of NN neurons are adjusted. This method allows us to approximate any problem to a highly complex mathematical function with a relatively small error rate [1].

Even though these hyper-parameterized non-linear models can establish a relationship between the input and the output of any complex system, the immediate structures of the model do not provide any insights into the relative importance, underlying relationships, structures of the predictors or covariates with the modeled outcomes [2]. Apart from this fact, from a traditional statistics viewpoint, an artificial Neural Network (ANN) is a non-identifiable model since two different NN trained by the same dataset can have a different structure, weights, and bias: and still output the same result. The combination of these two factors deems ANN as black-box models.

1.1. Background

Myelodysplastic Syndromes (MDS) are comprised of a group of morphological, immunophenotypic, functional, and genomic alterations in the hematopoietic linage. This group of clonal hematological diseases is characterized by ineffective hematopoiesis and recurrent genetic abnormalities such as cytopenias. In the long term, MDS can develop into Acute Myeloblastic Leukemia (AML) [3].

Diagnosis of MDS is supported by clinical information, blood count parameters, genetic studies, flow cytometry, and morphological abnormalities in Bone Marrow (BM) and Peripheral Blood (PB) cells. Since dysgranulopoiesis affecting neutrophils is more apparent in PB smear microscopic examination rather than in BM, pathologists rely on this relatively inexpensive, readily accessible tool to perform differential diagnosis of MDS. Nevertheless, pathologists still face difficulties when assessing dysgranulopoiesis, as cytoplastic hypogranularity is hard to quantify by the human eye, leading to dissonant prognosis amongst experts [4].

Within the framework of the diagnosis of these syndromes, A. Acevedo, as part of her Ph.D. dissertation [5] in collaboration with A. Merino, from the Haematology and Cytology Unit of the Hospital Clínic of Barcelona, developed DisplasiaNet. This relatively lightweight Convolutional Neural Network (CNN), which only requires 0.207 GFLOPs to perform a prediction, is able to recognize the presence of hypogranular dysplastic neutrophils in peripheral blood smear images with a global accuracy of 94.85%. This tool is intended to be used as a secondary diagnosis support tool when the pathologist suspects MDS based on other substantial evidence of these syndromes.

Although Artificial Neural Networks (ANN) have proven to be extremely powerful tools regarding image analysis and classification, a certain lack of trust has surrounded such technology, funded partially by the black-box nature in which they operate. This fact has become the primary obstacle for the acceptance of ANNs in mission-critical applications such as medical diagnostics.

One of the most contemporary lines of research to cope with this generated mistrust is Neural Network Interpretability. This line of research deals with developing new methods to visualize features and concepts that have been learned by a neural network, with the aim of being able to explain the underlying decision criteria that funds individual predictions.

Applying NN interpretability methods to DisplasiaNet is the next logical step towards integrating this handy tool into the pathologist's diagnostics workflow. Understanding the inner works of the NN will not only shed light on the decision criteria, but it will also bestow the pathologist and fulfill the ethical and legal requirements of contestability, following article 22. of the EU General Data Protection Regulation (GDPR) "right to explanation" [6].

1.2. Project objectives

This project aims to perform a Deep Learning Interpretability study on the Neural Network DisplasiaNet. This particular CNN detects whether a neutrophil cell image, acquired by microscopy, presents dysgranulopoiesis or not. Therefore, the objective is to prove that DisplasiaNet bases its decision criteria upon the same features as the expert pathologists.

By applying different interpretability methods, insights into the inner works of the CNN will become palpable. This acquired understanding must allow us to answer the following questions: Does the NN focus on the same features as the pathologist? Does the NN evaluate the features in the same matter? Is the NN able to generalize? Moreover, can the wrongly classified images provide information on how to improve the NN?

The answers to these questions contribute to deem the NN with Transparency, Explainability, and Reliability as the ultimate goal is to build enough trust to be able to integrate such a tool into the medical diagnosis workflow.

1.3. Project Scope

1.- The starting point of this project is the doctoral dissertation by Andrea Acevedo [5]. As such, the first step is to understand the architecture of the CNN, DisplasiaNet, as well as how it was trained and what data was used.

2.- The second step is to acquire knowledge on the disease, Dysplasia, that affects neutrophils. The aspect we are primarily interested in is the morphological features that expert Haematologists use to discern whether a cell presents dysplasia or not. Based on this knowledge, a python based online app is created to document the process of diagnosing a cell with dysplasia. In collaboration with Anna Merino from the Hospital Clínic de Barcelona, a carefully selected neutrophil cell image repository is diagnosed, annotating key areas on each image and scoring the different features.

3.- The third step is to apply different NN interpretability methods to the NN. Several aspects are to be analyzed. These include gaining insights into the different convolutional filters to understands which features the NN has learned to identify to further classify an image as normal or dysplastic (Class Activation Maps)—backpropagating the algorithm to identify what pixels of the image have been decisive in classifying the image (Saliency Maps). Occluding regions of the input image to locate what regions affect the classification the most (Occlusion Sensitivity Maps). As well as performing gradient ascent to locate critical features in the image during classification (Gradcam). Apart from these methods, images with the background cells removed will also be analyzed to prove the NN focuses on the neutrophil in order to perform the classification.

4- Finally, a comparison between the information gathered with annotation application from the expert pathologist and the information gathered from the interpretability methods is performed to establish a common ground for the decision criteria. This step will prove or refute if the NN can correctly classify Dysplastic Neutrophils and help convince the officials in charge to add DisplasiaNet as a support tool for the diagnostic of Myelodysplastic Syndromes(MDS)

1.4. Understanding Dysplasia in Neutrophils

The detection of Dysplasia in Neutrophil cells is an important indicator of MDS. As such, the morphology of a normal neutrophil and the morphology of a dysplastic Neutrophil will be overviewed. Most of each condition's characteristic features will be discussed to understand the procedure an expert pathologist uses to diagnose these cells visually.

1.4.1. What are the Myelodysplastic syndromes?

Myelodysplastic syndromes (MDS) are a group of disorders characterized by abnormal development of one or more of the hematopoietic lineages generated in the bone marrow. These are constituted by a variety of morphological, immunophenotypic functional, and genomic alterations. These alterations appear during the cell differentiation and maturation stage, classifying them as clonal hematologic diseases.

These syndromes have in common an ineffective Myelopoyesis, production of the Myeloid lineage cells (Figure 1), which lead to cytopenias, reduction in the number of mature blood cells, and medullary aplasia, failure of the bone marrow to function normally.

MDS are usually observed in patients over 50 with a typical onset age of 70 years [7] and more frequently amongst men. The symptoms include infections, bleeding and up to 30% of the cases turn into acute myeloblastic leukemia (AML) [3]. The primary etiology of MDS is unknown, although there is a genetic and environmental component linked to it, such as some hematological diseases (aplastic anemia, nocturnal paroxysmal hemoglobinuria, etc.), genetic disorders (Down syndrome, congenic disquerasitosis, etc.), exposition to toxic substances (benzene, metals, etc.), or specific treatments (chemotherapy, radiotherapy) [8].

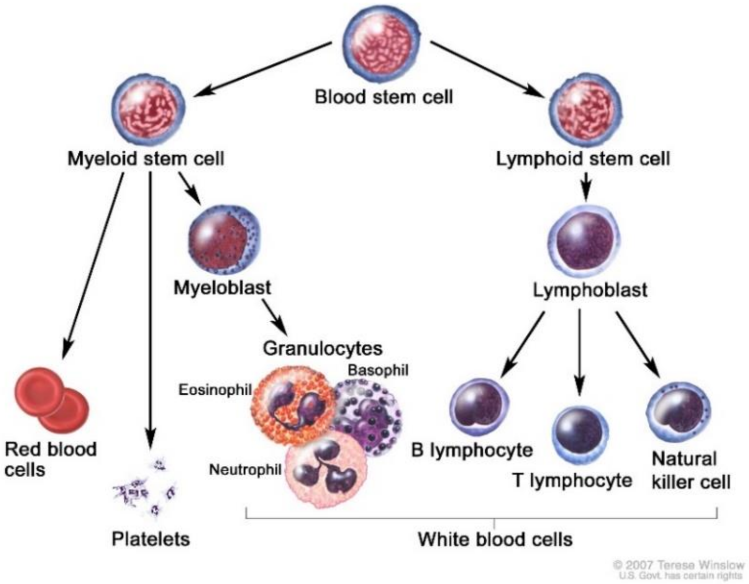


Figure 1. Hematopoietic cell classification source: [9]

To diagnose MDS, medical experts rely on differential diagnostics by comparing BM aspirate, BM biopsy, Blood tests, and the patients' medical records. Complete blood counts parameters, flow cytometry, genetic studies, BM morphology evaluation, and dysplasia in blood and bone marrow follow guidelines established in the WHO MDS 2016 classification [9]. Blood smear microscopic examination is a low-expensive and easily accessible tool for the assessment of cell abnormalities. It has a relevant role in supporting the initial MDS diagnosis since dysplasia is observed in one or more major myeloid lineages. Out of the three major myeloid lineages, dyserythropoietic and megakaryocyte, morphological alterations are more apparent in BM. Instead, dysgranulopoiesis, more apparent in PB.

1.4.2. The Neutrophil

Neutrophils along with Eosinophils and Basophils, constitute a group of Leukocytes (white blood cells) known as Granulocytes. These are mature blood cells derived from the Myeloid lineage of the Hematopoietic stem cells (blood stem cells) (Figure 1). Leukocytes are an integral part of the body's immune system, allowing it to fight infections and other diseases. Neutrophils are the most abundant type making up from 40 to 70 percent of them [10]. These are short-lived and are produced within the bone marrow through stimulation with granulocyte colony-stimulating factor. Unlike other Leukocytes, Neutrophils are not bounded to a specific circulation area; these have the capability of moving freely through the blood vessel walls into other tissues in the body to attack antigens. It is also worth noting that once the neutrophil has left the blood vessels, it does not return to them dying at the infection site, contributing to the formation of the whitish exudate called pus. [10]

1.4.3. Morphology of a Normal Neutrophil

Neutrophils are fairly uniform in size, with a diameter varying between 9 and 15 µm. Contrary to Erythrocytes (red blood cells) and Thrombocytes (platelets), Leukocytes are endowed with a nucleus. These have a single polymorphic-shaped nucleus consisting of two to five lobes chain-connected by a thin chromatin filament. This relatively small nucleus occupies about 21% of the cell's volume and is densely packed with chromatin which disappears as the cell reaches the end of life [11].

The cytoplasm of the neutrophils contains numerous granules with different morphologies. The primary granules, often larger and rounder in size, contain microbicidal agents. Other smaller granules, called secondary granules, house enzymes such as lysozyme, gelatinase, collagenase, and many others. These allow the cell to move fast through the different tissues as well as perform the basic functioning need for the cell to maintain alive. [12]

Once the neutrophil has been stained through the process of the May Grünwald-Giemsa protocol, the granules acquire a purple-to-lilac color, the cytoplasm acquires a pale pink color, and the nucleus acquires a deep blue-violet color. Upon further inspection under a microscope, the nucleus should present a homogeneous stain, primary granules should be a distinct Azure color, and Secondary granules should present a salmon pink color. Figure 2 contains a representative normal neutrophil used in the study.

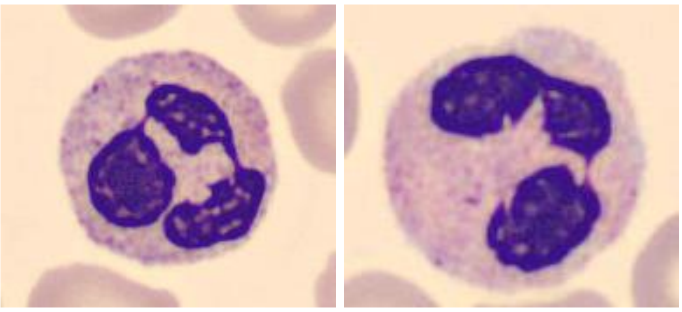


Figure 2.A. Healthy Neutrophil. (TN_SNE_14872673.jpg) B. Healthy Neutrophil. (SNE_131893). Source: [14]

1.4.4. Morphology of a Dysplastic Neutrophils

An unhealthy neutrophil can present several morphological features that pathologists use to determine dysplasia in the granulocytic lineage [8] [13]. These features can be presented in both the nucleus and the cytoplasm.

Cytoplasmic hypogranularity or agranularity is considered a particular dysplastic feature. It is represented by a reduction of at least 2/3 of the granules in the cytoplasm (Figure 3**Error! Reference source not found.**.A). Bear in mind that a subpar staining procedure might induce the agranularity of a sample ocasionally. Nonetheless, when it occurs, all the neutrophils present in the smear are hypogranular. Other dysplastic features that might be present in the cytoplasm are Dohlë bodies. These inclusions correspond to agglutinated ribosomes that present themselves as a light blue-colored region in the outer part of the cytoplasm of the neutrophil. It is shown in Figure 3.B and Figure 3.C

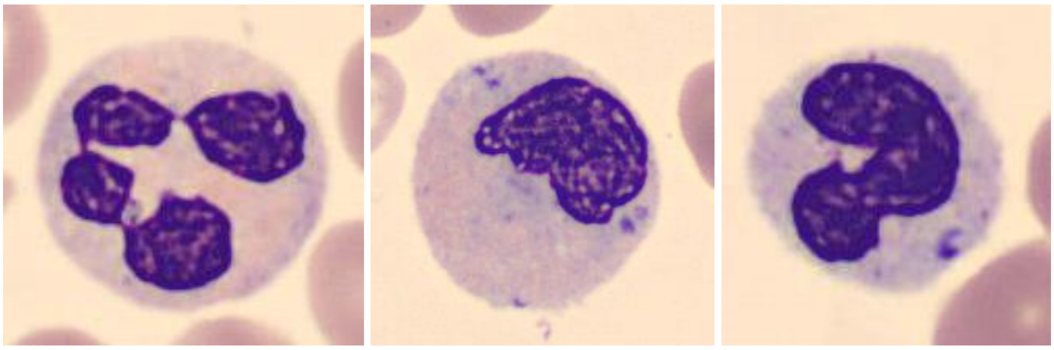


Figure 3.A. Agranular Cytoplasm (SNE_1901668.jpg) B. Dohle body (MMY_780452.jpg) C. Dohle Body (SNE_769084.jpg). Source: [14]

The most characteristic features related to the nucleus are linked to the lobe segmentation. Hiposegmentation is expressed when the nucleus presents less than three lobes. Often this feature is related to the pseudo-Pelger-Huët nucleus(Figure 4.A.), a bilobed nucleus shaped like a dumbbell and containing a coarse and lumpy structure. In other cases, the nucleus has not segmented at all and presents a chromatinic density proper to more mature neutrophils (Figure 4.B.).

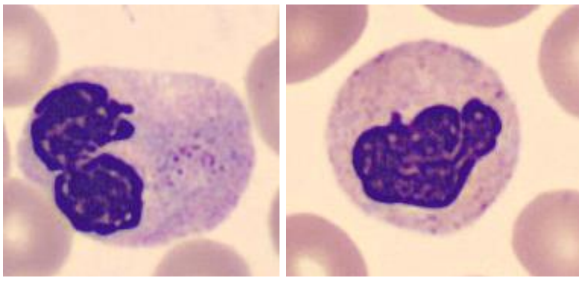


Figure 4.A.PseudoPelger-Huët Nucleus.(WD_SNE_2683958.jpg) B. Hiposegmented Nucleus (WD_SNE_2426196.jpg). Source: [14]

Another common nucleus dysplastic feature is Hipersegmentation, given when the nucleus contains more than five lobes (Figure 5.A), or "ring nucleus" (Figure 5.B). Band neutrophils are frequent in infections and may present themselves as hypogranular in MDS (Figure 5. C).

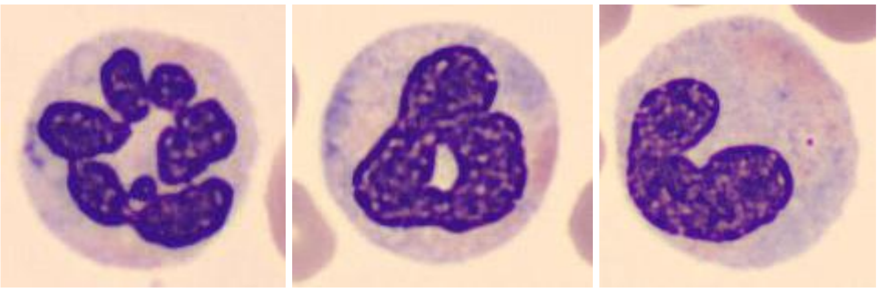


Figure 5. A.Hipersegmented Nucleus (SNE_1896407.jpg) B. Ring Nucleus (SNE_741882.jpg) C. Band Nucleus (BNE_1841329). Source: [14]

Other factors to consider when inspecting a neutrophil nucleus are the presence of nuclear appendixes, abnormal chromatin density, abnormal chromatin clumping, and abnormal segmentation patterns such as Mielocatexis.

Ultimately, the most important characteristics when diagnosing a cell will dysplasia are: Hipogranularity/Agranularity of the cytoplasm, Hiposegmentation/Hypersegmentation of the nucleus, and the chromatinic aspect of the nucleus Homogenic/Heterogenic.

1.5. Understanding DisplasiaNet

The following chapter aims to present the reader with knowledge on the object of study, the CNN model: DisplasiaNet.

DisplasiaNet is one of the contributions made by A. Acevedo's doctoral Thesis, Deep Learning System for the Automatic Classification of Normal and Dysplastic Peripheral Blood Cells as a Support Tool for the diagnosis [5]. This CNN has been developed with the aim of being integrated into the Hematological laboratory workflow as a support tool to classify PB Neutrophil images into dysplastic or normal.

Firstly, the dataset used to train, test, and evaluate the model will be overviewed. This section will detail the conformation of the dataset as well as some representative statistics. Secondly, the architecture of the CNN will be presented. This information will be expanded to contain conceptual knowledge on NN theory. To sum up, the final results and the criterion to interpret the model's output will be briefly discussed. This whole chapter is based on A. Acevedo’s doctoral thesis [5] and the resulting articles from it [14].

1.6.1. The database

A neural network is a series of complex algorithms that endeavor to recognize underlying relationships in a dataset. This dataset is the object of study, and by learning to identify and relate its underlying features enables us to make predictions on new images. As so, the dataset needs to be a representative sample of the data we want to predict. Larger datasets are an advantage as variability will be captured better. All datasets for image classification, like DisplasiaNet, must contain the study images and an additional annotation file stating what class the image belongs to.

The database used to train DisplasiaNet contains 20,670 images of neutrophil blood cells of which, 8,676 belong to dysplastic cells, and 11,994 belong to normal cells. All the images were gathered by the Hematological department at the Core Laboratory of Hospital Clinic of Barcelona.

A group of expert clinical pathologists selected and labeled the images according to their morphological characteristics. Patients' diagnostics (ground truth) were confirmed by integrating all supplementary test results: clinical data, PB and BM (Bone Marrow) morphology, flow cytometry, cytogenetics, and molecular biology following WHO 2016 classification of tumors of hematopoietic tissues [9].

The images were extracted from a total of 249 peripheral blood (PB) smears from 144 different patients. Peripheral blood is the blood that travels through the heart, arteries, capillaries, and veins. A blood smear is a blood sample, usually extracted from a vein, that is tested on a specially treated slide.

The process to obtain the images is as follows. The smears are collected in tubes containing Ethylene Diaminetetraacetic Acid (EDTA), an anticoagulant used to maintain the blood in a fluid state during hematological testing. The smears are then processed by the Sysmex (Kobe, Japan) SP-10i automatic slide maker and stainer. This automatic process ensures a constant preparation of slides and eliminates any human variability and error. The protocol used to stain the cells is the May Grünwald-Giemsa staining protocol (MGG). This staining process helps to visually differentiate the different blood and cellular components employing azurite dyes.

Each individual neutrophil image is automatically captured employing CellaVision®DM96 (CellaVision, Lund, Sweden), a powerful microscope that is able to locate and pre-classify blood cell types. The output are RGB images of 360x360 px. These images usually contain a single neutrophil cell located in the center that covers 16% of the image. The background usually contains several red cells and occasionally platelets (Figure 6).

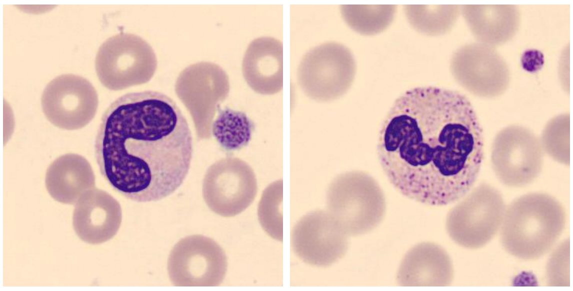


Figure 6. Examples of the dataset images: dysplastic cell(right), normal cell (left). The Neutrophil cell is in the center of the image. Red blood cells conform the majority of the background. Platelets can occasionally be observed, represented as smaller lilac colored isolated structures. Source: [14]

The dataset has been split into the training set (23.27%), the validation set (4.84%) and the test set on individual images(4.84%). Parallelly another test set was used to test 113 different blood smears of 72 patients (31.70%). The remaining 35.36% was used for the proof of concept by analyzing 116 complete blood smears from 72 different patients.

1.6.2. Architecture

DisplasiaNet is a Convolutional Neural Network (CNN). This type of artificial neural network has the capability of recognizing and classifying particular features from images and is currently one of the best performing architectures for image analysis.

DisplasiaNet presents the advantage of being a relatively lightweight model consisting of only 72,977 trainable parameters. Usually, other pre-trained image processing CNN such as VGG16 or AlexNet, contain 138.4 million and 62 million trainable parameters, respectively. The number of trainable parameters directly affects the computational power required to perform a forward pass measured in FLOPs (Floating Point OPerations). DisplasiaNet requires 0.207 GFLOPs as opposed to VGG16 that requires 15.3 GFLOPs, and AlexNet that requires 1.5 GFLOPs

A typical CCNs architecture contains a convolutional block followed by a dense block. The convolutional block is the one in charge of detecting and highlighting special features in the input image. The dense block performs an interpretation of the highlighted features and performs the actual classification.

Specifically, DisplasiaNet is conformed by four convolutional blocks, a flattening layer, and a Dense block containing two fully connected layers. The input layer accepts RGB images with a dimension of 299x299 px, and contains a total of 268203 neurons. The output layer contains one single neuron, which returns a value ranging from 0 to 1. This value can be interpreted as a probability distribution, where an output close to 0 indicates a higher chance of the image containing a dysplastic cell; otherwise, a value closer to 1 contains a normal cell. (Figure 7).

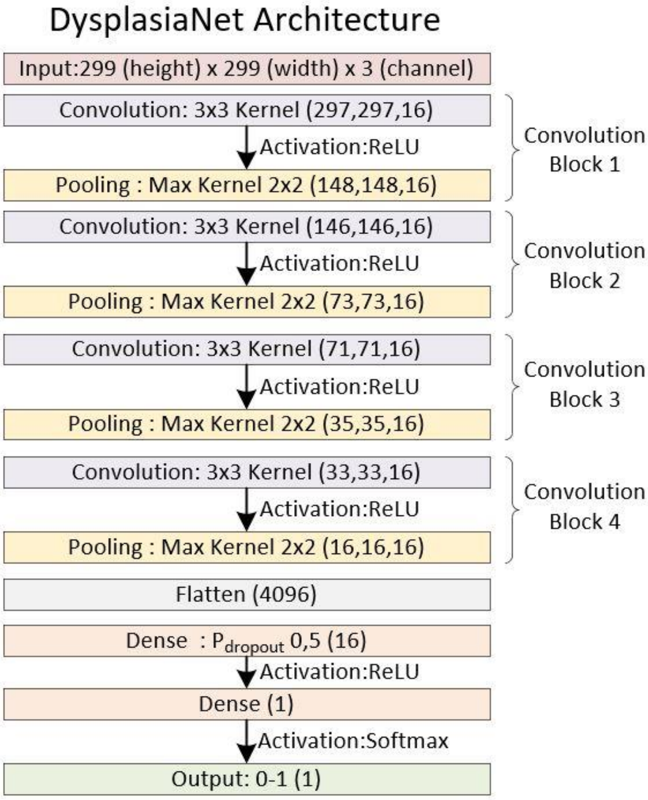

DysplasiaNet Architecture
Input:299 (height) x 299 (width) x 3 (channel)
Convolution: 3x3 Kernel (297,297,16)
Activation:ReLU
Pooling : Max Kernel 2x2 (148,148,16)
Convolution Block 1
Convolution: 3x3 Kernel (146,146,16)
Activation:ReLU
Pooling : Max Kernel 2x2 (73,73,16)
Convolution Block 2
Convolution: 3x3 Kernel (71,71,16)
Activation:ReLU
Pooling : Max Kernel 2x2 (35,35,16)
Convolution Block 3
Convolution: 3x3 Kernel (33,33,16)
Activation:ReLU
Pooling : Max Kernel 2x2 (16,16,16)
Convolution Block 4
Flatten (4096)
Dense : $P_{dropout}$ 0,5 (16)
Activation:ReLU
Dense (1)
Activation:Softmax
Output: 0-1 (1)

Figure 7. DisplasiaNet Architecture. Source:[3]

Each convolutional block is composed of a Convolutional Layer, an activation layer and a Max Pooling layer.

The convolutional layer of a CNN, named after the mathematical operation it performs (convolution), convolves the layer's input through the repeated application of the same filter resulting in an activation map. During the training stage of the NN, the filter is modified and adjusted automatically through the backward propagation operation, to highlight specific attributes in the image. For low-level convolutional layers these might be straight lines, circles, or spots; instead, for higher-level layers it might be able to detect textures and specific irregular geometries.

DisplasiaNet applies sixteen 3x3 px convolutional filters, or kernels, on each convolutional layer. The output of each layer are sixteen specific feature maps. Each one highlights the specific feature each filter has specialized in. Also, as a consequence of the convolutional operation, the output is 2 px smaller than the input on each axis. This information is important as it will be needed to properly overlap the feature maps on the input images later on during the analysis.

Each feature map is then evaluated with the Rectified Linear Unit (ReLU) activation function (Eq. 1). This piece-wise function returns the input when it contains positive values and a zero when it contains a negative value. The resulting feature map will only contain positive values, therefore only accentuating features that react positively to the filter. The application of this activation function is extensively used in CNN architecture and helps to speed up the learning process.

$$f(x) = x^{+} = Max(0, x) \qquad \textit{(Eq. 1)}$$

The pooling layer passes a two-dimensional kernel over each channel of the feature map to summarize the features lying within the region covered by the filter. This action decreases the size of the feature maps and, as a consequence, reduces the computational cost.It also has the added benefit that it creates invariance amongst the image and produces features that are more robust against noise. In DisplasiaNet, the kernel is a 2x2px max pooling operation. Out of the many pooling operations, MaxPooling selects the maximum pixel value of the region located in the kernel to pass as the region's output value. The operation reduces the output of the layer dimension by the factor of the kernel used.

Overall, the convolutional blocks reduce the RGB input image into sixteen 16x16px feature maps. For this information to be processed by the dense block, the different feature maps must be combined into 1-D vector. This is achieved by the flattening layer that accepts an array of (16,16,16) and outputs a 1-D vector containing 4096 values.

The last part of the model is the Dense block which is formed by two fully connected layers. These types of layers are the most common and frequently used in deep learning applications. All the neurons of the layer are interconnected with all the outputs of the previous layer. The dense layer performs a matrix-vector multiplication and returns an 'm' dimensional vector as a result. Due to its properties, this layer is commonly used to change the dimensions of a vector, like in DisplasiaNet. It is also used on rotation translation and scaling operations.

The first FC layer has 16 neurons and has the particularity of having been trained with a neuron dropout chance of p_{drop}=0.5. This means that at each training epoch, neurons have a 50% chance of being randomly deactivated. This approach prevents the model from overfitting and yields better results as it teaches the NN to generalize by not focusing on specific patterns. The output of each of the 16 nodes the is evaluated with the ReLU activation function to prevent passing on negative values.

The final layer only has one node. The 16 inputs of the neuron are weighted and evaluated with the sigmoid activation function (Eq. 2). This activation function is commonly used for binary classification models as the output is comprised between 0 and 1. The output represents a probability distribution, where values closer to zero represent a high chance of the image containing a dysplastic Neutrophil and values closer to 1 a normal neutrophil.

$$f(x) = \frac{1}{1 + e^{-x}} \qquad \text{(Eq. 2)}$$

The final model is a relatively lightweight model containing only 72,977 parameters. This feature allows the model to be deployed onto low-power devices and speeds up the smear analysis process.

1.6.3. Evaluation and Result Interpretation

This architecture was the best performing out of a 10 model set that varied the number of convolutional blocks and fully connected layers. The proof of concept test dataset is comprised of 7308 images from 66 smears of 32 patients suffering from MDS and 70 smears from a 40 healthy patient control group. When DisplasiaNet was evaluated with this dataset it achieved a global accuracy of 94.85 %, a sensitivity of 95.5 %, a specificity of 94.3 %, and a precision of 94 %. Accuracy describes how many images out of the total were correctly classified. Sensitivity describes how many dysplastic neutrophils are detected from the total of dysplastic images. Specificity describes how many normal cells are detected from the total normal images. Specificity describes the total of dysplastic cells that are correctly classified out of the correctly classified images.

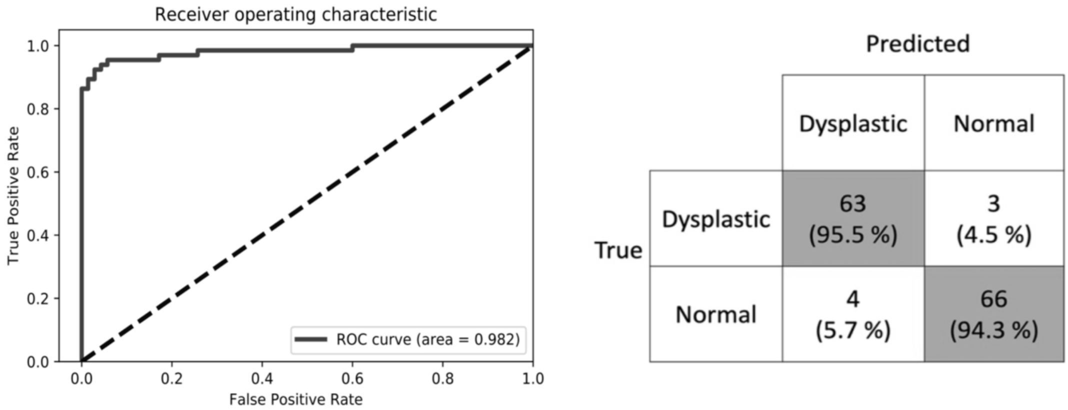
IMAGE SUBSETS

		Training		Test by cell	Test by smear	Proof of concept
		Train	Validation			
CELL IMAGES	Dysplastic	1,887	500	500	2,718 (49 smears)	3,071 (66 smears)
	Normal	2,923	500	500	3834 (64 smears)	4237 (70 smears)
TOTAL IMAGES		**5,810**		**1,000**	**6552** (113 smears from 72 patients)	**7,308** (136 smears from 72 patients)

Table 1. Dataset for training and testing DysplasiaNet CNN. source[3]

As detailed in the previous section, the model outputs a value between 0 and 1, representing a probabilistic distribution. Values closer to zero represent a high chance of the image containing a dysplastic neutrophil, and values closer to 1 a normal neutrophil. However, the result must be simplified to a binary value. To obtain the optimal threshold value, the original paper performed a

ROC curve (receiver operating characteristic curve) analysis. This graph presents the trade-off between sensitivity and specificity. Classifiers that give curves closer to the top-left corner indicate a better performance. As a baseline, a random classifier is expected to give points lying along the diagonal. The optimal cut-off value is 0.123. This value corresponds to a threshold of 0.979. Any value below this threshold will be considered dysplastic and above it normal.



		Predicted	
		Dysplastic	Normal
True	Dysplastic	63 (95.5 %)	3 (4.5 %)
	Normal	4 (5.7 %)	66 (94.3 %)

Figure 8. ROC curve analysis and truth table for the Proof of concept dataset. Source: [3]

1.7. Understanding Neural Network Interpretability

This section introduces Neural Network Interpretability by first explaining what it is, why it is needed, and what output it is expected to deliver. Secondly, a systematic review of published papers will be performed. In it, a full taxonomy of interpretability techniques will be performed. This state-of-the-art review will also discuss some medical-related papers where interpretability techniques have been applied.

1.7.1. Why is Neural Network interpretability needed?

Deep Learning has become an indispensable tool for many scientists tackling complex problems. These networks have been trained to perform a wide variety of applications ranging from captioning and classifying images, speech recognition, translating text, and even predicting events from a collection of data. In many cases, this technology has enabled the automation of previously reserved tasks for humans, often outperforming in precision and speed terms. Although the application of DL has become a ubiquitous solution to many real-world problems, our understanding of why they are so effective is lacking. This fact is attributed to the black-box nature of DL, in the sense that although it performs very well in practice, it is difficult to justify the underlying mechanisms and behaviors.

For many cases, these empirical results should not be possible according to sample complexity in statistics and nonconvex optimization theory, although insights are being found in the geometry of highly-dimensional spaces [15]. Questions regarding the inner workings on how the models learn to make accurate predictions, why some features are favored by others in the models, and what changes can be made to improve the outcome, are often arised. Unfortunately, only a mild success has been made towards answering these questions

To open up the black-box nature of CNN's many researchers have turned towards model interpretability. In layman's terms, Interpretability is defined as the ability to provide explanations in understandable terms to a human. However, we must bear in mind that the term interpretability is formally ill-defined, as there is a lack of consensus among experts as motivations and definitions for the term differ and are even dissonant between papers [16]. Nonetheless, the idea of “you will know when you see it” is present in most papers [17].

Albeit, some papers [18] [19] differentiate between transparency, interpretability, and explainability. In a broad sense, these consider that transparency is linked to the ML approach, interpretability to the model in combination with the data, and explainability to the model, the data, and the human involvement. For ease of explanation, we consider the three terms interchangeable as [2] does and not emphasize the subtle differences.

We must bear in mind the concept of DL interpretability is fairly recent. The survey by F.-L. Fan *et al.* [20] searched for the terms "Deep Learning interpretability", "Neural Network interpretability", "Explainable Deep Learning", and "Explainable Neural Network" on the web of science as well as Google Scholar, PubMed, and other scientifical journeys for the time period of the year 2000 to 2020. The search returned that prior to 2015, the average published papers related to this topic were less than forty per year and only contained the term NN interpretability. After 2015, the different terms experienced an enormous boom, as publications related to these terms underwent exponential growth.

Usually, when a NN is trained, its performance is evaluated with regard to high accuracy. This metric, however, is not enough to describe real-world tasks as stated by F. Doshi-Velez and B. Kim [21]. The real world is highly dimensional, and there may not be any low-dimensional model that can be fit to it. NN Interpretability is essential in many sensitive fields, such as the medical, where the models are required to hold accountability.

Although we are currently in the big data era, high-quality data is not always accessible. ML models tend to pick up biases from the training data by default, as that is how the reality of the problem has been presented. Although the process of training and validating a NN model already accounts for data variability, a shift in the distribution data may occur when the model has been deployed. Phenomena described by Goodhart's law, specification gaming, or failure to generalize features can lead to disastrous predictions. Additionally, A. Nguyen *et al.* [22] demonstrated that unperceivable changes in the pixels of the input image could drastically change the score of the output prediction.

As a result, a need to prove the underlying decisions mechanisms of NN is generated. Firstly, NN interpretability can aid users in gaining trust. Humans tend to be reluctant to rely on new technologies for critical tasks unless we have a deep understanding of how it works. If a model builder can argue the reason a particular model makes a specific prediction under a controlled environment, users would know whether such model contributes to an adverse event or not. Focusing on transparency can aid in mitigating such fears.

Secondly, safety must be ensured. High-reliability requirements are demanded, and real-world environments are complex. DL is, to an extent, a collection of highly non-linear algorithms. The mathematical operations DL is built on finds relations between variables in a latent space. Interpretability techniques can highlight the features the NN considers relevant, allowing us to evaluate if the decision criteria are correct. Moreover, these techniques can also aid in pinpointing the root issue and allow us to provide a fix accordingly. Ambiguously, interpretability does not improve the performance of a NN or make it more reliable; however, it is an intrinsic component for a highly reliable system.

Lastly, there is an ethical and legal requirement for model interpretability. In the first instance, a neural network model should avoid algorithmic discrimination. This is particularly relevant in areas such as credit risk assessment, mortgage qualification, and hospital readmission prognosis. The USA's Food and Drug Administration (FDA) requires interpretability for any new drug developed utilizing ML in conjunction with clinical test results [23]. The same applies to any medical device.

In 2016 the European Parliament passed regulation 2016/679 in relation to the EU General Data Protection Regulation [24]. In paper [6], the potential impact of this regulation is summarized and concluded with the "right to explanation" that citizens must have. This encompasses the right to be given an explanation for the output of any algorithm that can impact individual rights. As such, contestability must be ensured. Given the case of NN, these do not return predictions with o decomposition on the desition criteria. As such, in medical applications where the patient's well-being is at risk, Pathologists using this technology must be able to argue their actions and justify the result provided by the NN. Interpretability techniques that are able to il·lustrate a chain of reasoning can help with such appeals.

A good interpretability model or a good explanation format can vary as people seldom require complete sentences to understand a concept. This opens a window of "explanatory languages," whether it be a natural language, logic rules, or even figures and diagrams. However, these explanations must follow the general guidelines of Exactness, Consistency, Completeness, Universality, and Reward [20].

Exactness refers to how accurate an interpretation is, whether it is limited to a qualitative description or it is a quantitative analysis. Quantitative analyses are often preferred as the relation between variables is numerically defined. Nonetheless, in many cases, this simplification is hard to achieve. This is the case for image analysis, where the variables are not directly palpable, and features are described by groups of pixels.

Consistency refers to the uniformity between explanations in the sense that there are no contradictions. In Interpretability, the No-Free-Lunch theorem applies, meaning there is not a universal technique for each type of problem. This implies that several interpretability techniques must be contrasted to achieve an overall picture of the decision criteria of the model.

Completeness considers how well the explanation fits the data. From a mathematical perspective, a NN learns to maps the input to the output in the best way possible. A reasonable interpretation must be applicable to the maximum number of data instances.

Universality describes, in a sense, the rapid growth of the interpretability knowledge domain. Finding a universal interpretation that deciphers as many models as possible can ease up the task of explaining the decision on criteria of the model by saving time and labor.

Last but not least Reward. The interpretability methods must be able to provide gains to the understanding of the network. In addition to aiding users' trust, the results should be report improvements, where allowed, to the architecture or training of the model.

2. State of the Art

ML interpretability has gathered much attention through the past couple of years due to the increased need to prove their underlying decision criteria. As such, many interpretability methods have been developed to cope with this need. It is important to notice that different types of input data require different methods to make sense out of the NN.

This section is divided into two main parts. A first part where different interpretability techniques are discussed, and a second part that discusses the implementation of these techniques in vanguard applications. All the research stated will be carefully tailored to suit the application of the problem stated in this project.

2.1. Interpretability techniques

Being NN interpretability a fairly recent domain knowledge, experts have not reached a consensus regarding a clear taxonomy to properly classify DL interpretability techniques. The most common classification are: Pre-model vs. In-model vs. Post-model; Intrinsic vs. Post-Hoc; Model-Specific vs. Model Agnostic; and Result of explanation methods. However, the chosen classification for this section will be Intrinsic vs Post-hoc, as it is the one that relates the most to the problem stated in this project. Nonetheless, the Model specific vs the Model agnostic features of each technique will be discussed as it is important to consider the implications of each type.

Post-hoc interpretability techniques are those that are applied to a model after this one has properly been trained. These techniques do not compromise the precision of the model since the prediction, and the interpretability are two independent processes without mutual interference. Nonetheless, no model can represent with 100% accuracy any other model as interpretation would become the original model. This fact attributes these techniques to be slightly inaccurate and require further comprehension. Despite their slight inaccuracy, they are also the easiest to interpret.

On the contrary, Intrinsic models eliminate the bias of the post-hoc techniques. This is since these techniques must be considered in the architecture development phase. These models restrict the complexity of the model and are considered interpretable due to their simple structure. This trade-off is usually represented in a loss of accuracy. Examples of these techniques are Liner regressions, Logistic regressions, Decision Trees [25], k-nearest neighbor [26], and the Naïve Bayes Classifier

[27]. Due that these techniques do not apply to our current model, no further explanation will be made.

2.1.1. Feature Analysis

Feature Analysis interpretability techniques focus their analysis on visualizing and comparing the different structures the model is composed of. Through the process of visualizing layer outputs and specific neurons, sensitive features and the ways the model processes data can be explained to some extend. These techniques provide qualitative insights into the NN in following a model agnostic approach. Nonetheless, these techniques lack an in-depth, rigorous, and undefined understanding.

The most straightforward approach for models that handle images is the method described by F.Chollet in [28]. This method, Intermediate Feature Maps, aims to visualize the internal features of CNN in a global matter. The output values for the different convolutional, pooling and activation layers can be obtained by feeding an input image to the model. As convolutional networks maintain spatial information, these outputs can be arranged to match the structure of the input image. As a result, a Feature map is obtained for each filter in the convolutional layer. These maps, usually represented as a heat map, highlight the structures of the input image that the filter considers to be relevant. From this method, we can extract knowledge on what image-specific features the model is considering to perform its prediction.

Numerous methods, [29] [30] [31], explore different strategies of generating synthetic input images devoted to maximizing the output of a NN or target structures of the model. The resulting images are popularly called 'deep dreams' as these depict what the 'dream' on the NN is supposed to be. In this image key structures for the different classes can be identified.

Another approach is the one by M.D. Zeiler *et al.* [32]. This paper describes the design of a deconvolution network consisting of unpooling, rectifying, and deconvolutional operations to be paired with the original convolutional network. This method aims to reconstruct an image from the information stored in the convolutional layers without purposely retraining the network. The reconstructed image allows identifying what features are being removed by the original convolutional model compared to the original input image.

2.1.2. Model Inspection

Model Inspection works in a similar mater to feature analysis inspection; however, these methods apply external algorithms to dive into the NN inner workings to extract important structural and parametric information. The application of analytical tools such as statistics deems these methods more trustworthy and rewarding.

A first initial approach is CAM [33] (Class Activation Mapping). This method bases its explanation on the weights of each filter on the last convolutional layer adopt towards an individual prediction. This method requires modifying the network by removing the fully connected layers (FC) present at the end of classic CNN architectures and substituting and training it with one FC layer with the softmax activation function. The output of the different filters on the last convolutional block are weighted and added, and finally overlaid on the input image. This highlights the relevant features detected by the convolutional blocks and how they affect the prediction. Grad-CAM [34] was posteriorly developed by combining the class conditional property of CAM with pixel-based gradient backpropagation returning better results. However, this method lacked stability, as when several occurrences of a class were present, its performance dropped, and single for object images, the method failed to highlight the whole region on the heat map. Grad-CAM++ [35] remedies the flaws as mentioned above of its predecessor by adding a pixel-wise weighting distribution on the final convolutional feature map. Finally, ScoreCAM [36] achieves a better visual performance with less noise and offers better stability than Grad-Cam and Grad-CAM++. This is achieved by replacing the gradient-based weight with a score-based weight that has a much smoother curve of change.

An alternative numerical approach is presented by A. Bansal *et al.* [37] . This paper developed a model agnostic algorithm to identify which instances a neural network is likely to fail to provide a prediction for. This model is to be executed in parallel with the original model. In such scenario, the output of the supporting model would activate when a potential misclassification image is being processed.

Y.wang *et al.* [38] proposed a method of demystifying models by analyzing critical data route paths for different class images. This method would allow to rout each image in the dataset and compare common features amongst them.

2.1.3. Saliency Maps

Another popular approach for interpreting image classifiers are Saliency Maps. This method is derived from the concept of saliency in images which depicts unique features, such as pixels, of the image which are relevant in the context of visual processing. As such, the output image of this method is a masked with the most relevant pixels for the classification highlighted.

The original method, Vanilla Saliency [39], evaluates the gradient of the class score as a function with respect to the input. In other words, it interprets the gradient as a sensitivity map. The basic concept is to first forward pass the input image through the whole model. The obtained data is then backpropagated all the way to the input data. The usual backpropagating operation performed during training stops at the second layer because there is no interest in changing the input data. As the backpropagation is carried all the way to the input, pixels are highlighted on the input image. The gradient is, in its essence, a list of derivatives, one for each pixel.

Smoothgrad [40] is an improved method to generate saliency maps. Its principle relies on the same as Vanilla Saliency; however, several input images are generated by altering the image with Gaussian noise and smoothing filters. Each generated map is feed into the model and backpropagated to the input. The final result is the average of the different obtained maps. This method has been proven to have a denoising effect on the final sensitivity map offering a better visualization.

2.1.4. Occlusion oriented Maps

Sensitivity Maps can also be generated in a model agnostic matter. These are based on the principle of occluding a part of the input image and evaluating the relevance of that region to the classification via the output score.

Occlusion sensitivity maps [32], are one of the first proposed methods. The principle relies on shifting a grey patch of fixed dimensions along the two dimensions of the image, occluding one region at a time. The output score of each of the analyzed images is plotted on a map where the patch has been located. The complete obtained map is a heat map, where high values contribute to a positive classification of the specific class. On the other hand, low values are attributed to features that, when occluded, negatively affect the prediction. To analyze this method, it is necessary to know what the benchmarked score of the image is to have a notion of what affects the model positively and what affects the model negatively.

LIME (Local interpretable model-agnostic explanations) [41], approaches this concept in a slightly different matter. The input image is previously segmented to create superpixels. These superpixels are group real image pixels by neighboring colors and structures, obtaining semantic image structures. The image is then iterated several times with a combination of different superpixels blacked out. Ideally, every superpixel combination should be processed; nonetheless, this computation would be highly costly. The class output score of each iteration for each pixel is fitted to a linear regression. The top superpixels of the regression should correspond to the most relevant regions of the image. Bear in mind that LIME only evaluates local accuracy. The combination of different features supports predictions in models.

SHAP (Shapley Additive exPlanations) [42] is another approach to analyze superpixel-based occlusion. This not only evaluates the local accuracy of the presence of features but also evaluates the missingness of the features and the consistency throughout the iterations. This analysis is achieved by the incorporation of Shapley values [43], derived from game theory, to decide the importance of the different segmented features.

2.1.5. Proxy Methods

Proxy methods construct a more interpretable simpler proxy that highly resembles the trained, large and complex original model but without the black-box nature. This is achieved by the integration of white-box models such as decision trees and rule decisions. These methods can return predictions in the local space or global solution space. However, the main drawback to this method is that an extra cost is needed to construct this alternate model.

2.2. NN interpretability in Medical Applications

The techniques mentioned above are currently being used to study the behavior of different CNNs. The output of the study is employed in different matters. Some are used to highlight features on the input image for the expert pathologist to have more information. In other cases, the information is used to learn more about a particular pathology as the CNN learns features that might be unperceivable by the human eye. Other uses imply learning the most relevant features of a CNN to train a simpler, less computing demanding NN. Some examples are listed below.

In P. Van Molle *et al.* [44] a CNN was trained to classify skin lesions. The feature maps from last two convolutional layers were extracted for each input Image, subsequently being analysed and compared. From this analysis, rules were extracted to aid the pathologists in the decision-making for skin lesion classification. Skin color, skin types, and the border of the lesion responded to different activation maps.

In A. Zaritsky *et al.* [45] a trained convolutional Encoder-Decoder NN generates artificial, exaggerated images of metastatic melanomas. With the use of NN interpretability methods, underlying features that can escape the human eye are learned by the pathologists to understand better how to diagnose these melanomas.

J.R Zech *et al.* [46] trained a model to classify patients, based on chest radiographies, into having pneumonia or not. To justify the desition making of the models, different correct and incorrect images were analyzed employing CAM. The results reported that the model occasionally focused on irrelevant features for the disease, such as metal tokens.

In Cruz-Roa *et al.* [47] a CNN is developed to classify if Basal cells presented Carcinoma Cancer. As a peculiarity of the architecture, the model had two outputs, one scoring the probability of the cell presenting cancer and the other not. This fact contributes in a positive way to the interpretability of the model. The authors also highlight the regions that made the model decide on a prediction on the final image (blue for Cancer and red for normal).

In S. Pereira *et al.* [48] brain tumor segmentation CNN was analyzed employing LIME. This application cast local interpretation efforts of the model.

In J. Diao *et al.* [49] a CNN is trained to detect five different types of cancer from different tissue types. From this information, an interpretability study is performed to obtained Human Interpretable Features (HIF). These obtained features are then used to train a simpler Dense NN used by the pathologist to aid in diagnosing the different tissue cancer.

3. Methodology

This chapter discusses the methods employed for performing the NN Interpretability Study. Aspects such as the election of the dataset and the theoretical background of the methods are discussed.

3.1. Interpretability study dataset

The first step towards performing a NN interpretability study is the conformation of the dataset. This dataset must be a subset of the images used to train the original CNN, DisplasiaNet [14], as these will tend to better represent the features that the NN has learned to make its predictions. These images are 360x360px in dimension and RGB format, meaning each image can be expressed as a three-dimensional array expressed as $i_m = (3{,}360{,}360)$. However, these images need to be down-sampled to $i_m = (3{,}299{,}299)$ to match the input layer of DisplaciaNet.

As for the images, these need to contain an equal amount of Dysplastic and Normal cells and be a representative sample of the dataset. This means that the selected images must contain: a wide variety of cytoplasmic granulation, varying from agranular to highly granular; a wide variety of nucleic chromatinic densities, varying from heterogenous to homogenous; and different segmentation stages, varying from hypolobulated to hyperlobulated. Other features must also be present in the dataset, such as Dohlë bodies and pseudo-Pelger-Huët nucleus. The meaning of the different features is explained in chapter 1.4, Understanding Dysplasia in Neutrophils.

For the images to be as representative as possible, these must be selected alongside the expert pathologist. We must keep in mind that many of the interpretability techniques applied to the image dataset must be analyzed qualitatively. Consequently, employing a small dataset is key to interpreting the output maps efficiently; therefore, the established number of images to be analyzed per class is 23. These contain the 3 images used as examples in the original paper of DisplasiaNet and twenty additional images that represent the class appropriately. Additionally, 5 misclassified images per class are selected to further understand the rationale behind the misclassification. The total amount of images in the dataset is 56 (Table 2). The different images selected have been coded by adding a suffix to the original image name, these are: Thesis Normal cell(TN), Thesis Dysplastic cell (TD), additional Normal cell (ZN), additional Dysplastic cell (ZN), misclassified Normal Cells (WN) and misclassified Dysplastic cell (WD)

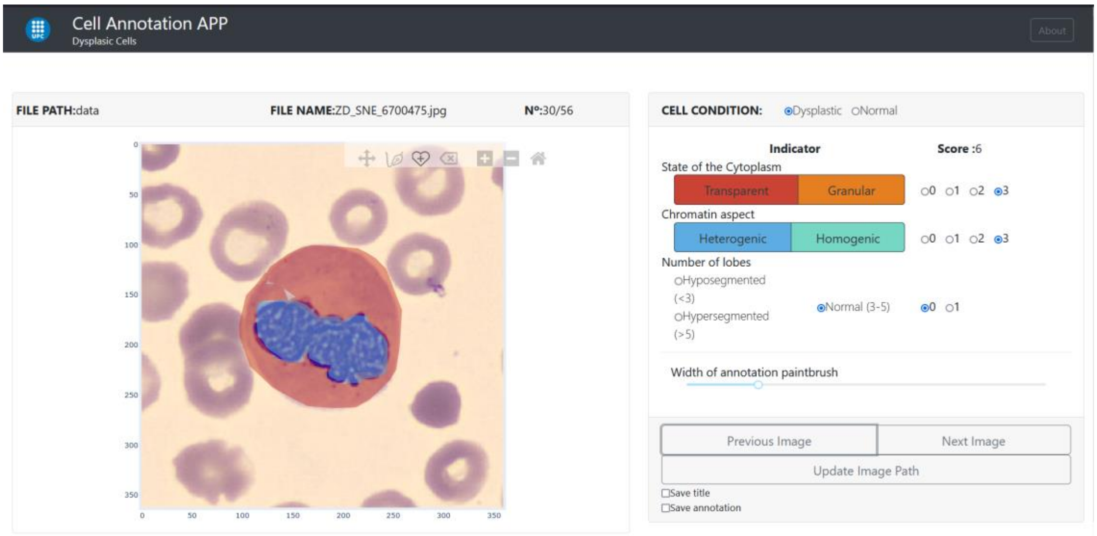
	Normal	*Dysplastic*	*Total*
Correctly classified	23	23	46
Incorrectly classified	5	5	10
Total	28	28	56

Table 2. Interpretability image dataset.

In order to obtain a baseline for comparison, all the images are processed by DisplasiaNet to obtain the output score.

3.2. Cell annotation APP

An online web annotation application has been developed to gain a deeper understanding of the workflow the expert pathologist follows to diagnose a Neutrophil with Dysplasia (Figure 9. Cell Annotation APP GUI). This tool enables us to thoroughly document and evaluate the key features of all the images contained in the Interpretability Dataset.



FILE PATH:data **FILE NAME:**ZD_SNE_6700475.jpg **Nº:**30/56

CELL CONDITION: ● Dysplastic ○ Normal

Indicator **Score :**6

State of the Cytoplasm: Transparent | Granular ○0 ○1 ○2 ●3

Chromatin aspect: Heterogenic | Homogenic ○0 ○1 ○2 ●3

Number of lobes: ○Hyposegmented (<3) ○Hypersegmented (>5) ●Normal (3-5) ●0 ○1

Width of annotation paintbrush

Figure 9. Cell Annotation APP GUI

In collaboration with the expert pathologist from the Hospital Clinic de Barcelona, and contrasting with the literature review, the three most important features to diagnose a Dysplastic neutrophil have been selected. The following are listed in order of priority: presence/absence of granules in the cytoplasm; the state of how the chromatin is presented in the nucleus; and the number of lobes the nucleus presents. To document how predominant a feature presents itself, a scoring system for each of the features has been developed. This scoring system is not based on any existing scoring systems and is merely a scale used to quantify the presence of each feature for each image.

Regarding the granularity in the cytoplasm, Normal cells typically present a high granule count, and Dysplastic cells typically present a low or occasionally a null granule count. This feature (S_g) is evaluated from -3 for highly granulated to +3 severely hypogranular.

As for the chromatinic state of the nucleus, normal cells typically present a homogenous tinction pattern indicating a highly dense nucleus. On the other side, Dysplastic cells typically present heterogenic tinction of the nucleus, indicating a low chromatinic density. This feature (S_c) is evaluated from -3 for highly homogenous tinction to +3 for a severely heterogenic tinction.

Lastly, the lobular segmentation is evaluated on a binary scale (S_l). A score of zero is given if the lobular segmentation is considered normal, between three and five lobules in the nucleus. A score of one is given when the cell presents an abnormal lobular segmentation, hypolobulated (<3 lobes), or hyperlobulated (>5 lobes).

Following this criterion, each image of the database should be assigned a total score (S_t) between -6 and +7, which corresponds to the addition of the independent feature scoring system $(S_t = S_g + S_c + S_l)$. Images with a total score higher than $S_t \geq 1$ indicate that the image contains a Dysplastic Cell and scores lower than $S_t \leq 1$ indicates that the image contains a normal cell.

Additionally, the application allows the user to manually segment both the cytoplasm and the nucleus by selecting the freehand drawing tool and tracing over the perimeter of both structures. This segmentation is useful as it can be converted into two independent masks to either isolate the cytoplasm or the nucleus. The application saves an annotation file for each image in *.json format that contains the different independent scores as well as the masks for both the cytoplasm and the nucleus.

Cell Annotation APP has been developed using the Plotly/Dash [50] productive web-based python framework, licensed under the MIT License. This framework has enabled us to create a user-friendly application for expert pathologists to annotate the different cell images in a simple, visual and fast matter.

The application has been packed into a Docker Image [51] and is licensed under a Docker free license for Education and learning and Non-commercial open-source projects. The Docker platform enables us to create virtual environments where to install different as well as their dependencies and pack them in the form of an image for easy deployment on a host computer. The docker container is currently deployed on the online server DeepBox located at the EEBE campus of UPC for online access.

3.3. Feature Maps

Out of all the different types of DL architectures, CNN are amongst the easiest to interpret in human terms. This is partially due to the architecture these models are based on: convolutional layers followed by an activation layer and a pooling layer. These layers are highly successful at maintaining the spatial information of the input image. The convolutional operation applies a kernel, also known as a filter, specialized in detecting specific patterns during the training phase of the model. Through the convolutional process of applying the filter to the input image, different features are highlighted. Typically, a convolutional layer is composed of several different filters, each one specialized in highlighting different features of the input. The different outputs for each convolution layer are known as a Feature Map or an Activation Map.

Feature maps provide insightful information on how successive convolutional layers transform their input. These pseudo images give a visual representation of what the different filters have learned to highlight. The output of a convolutional layer (A_i) can be understood as a three-dimensional image where the first and second dimension are the width (W_{conv}) and height (H_{conv}) of the feature maps and the third dimension is defined by the number of channels/filters the convolutional layer is composed of $(A_i = (Nº_{filters}, W_{conv}, H_{conv}))$. As each channel of the convolutional layer encodes relatively independent features, the proper way to visualize these feature maps is by independently plotting each channel as an individual 2D image. Figure 10 shows all the feature maps that can be extracted from DisplasiaNet for a specific input image.

When successive convolutional blocks are stacked, the output of the previous block becomes the input of the successive block. Due to the pooling operation, each successive block outputs a smaller feature map. This reduction factor is directly related to the pooling kernel dimension.

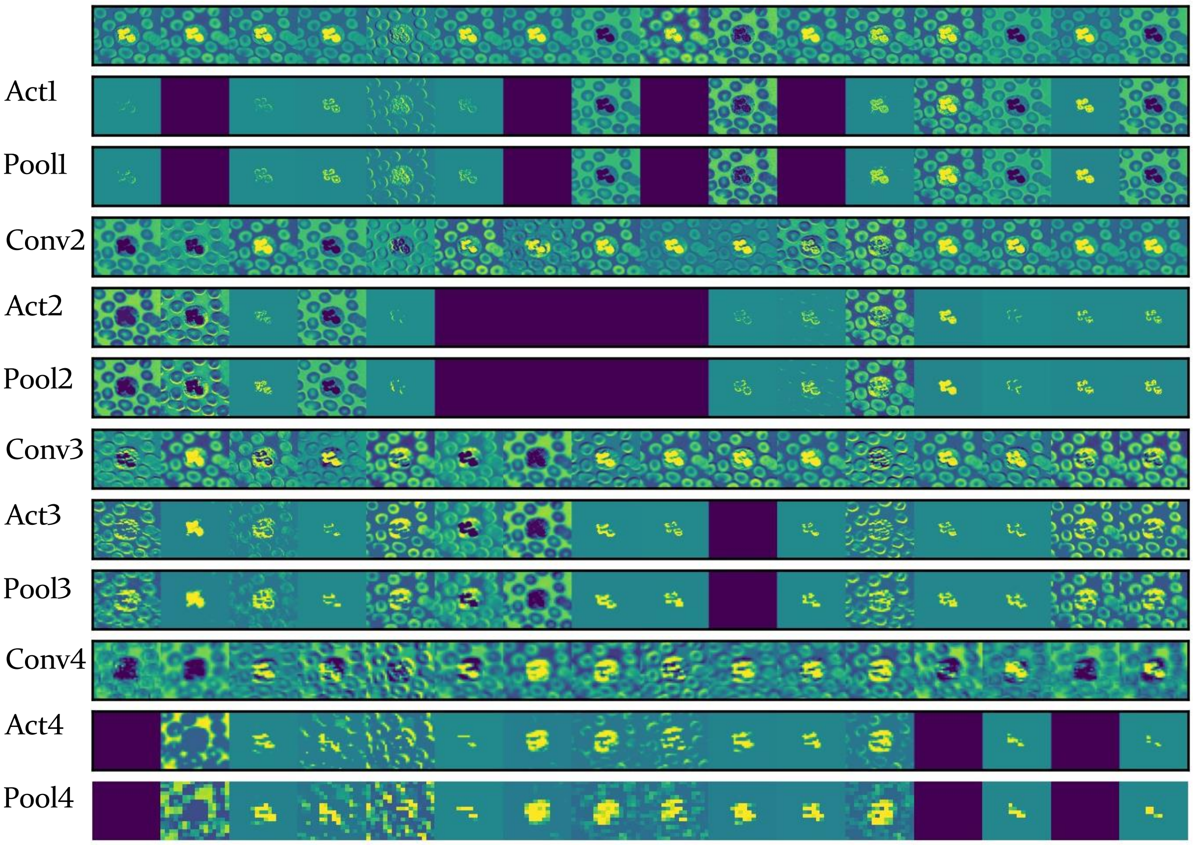


Figure 10. Feature Maps of a Normal Cell (TN_SNE_118039.jpg)

Arguably the first convolutional layer's filters act as a collection of rather simple edge detectors, retaining almost all the information present on the input image, as the layers go deeper, the output feature maps become increasingly abstract and less visually interpretable. These contain less and less information about the input specific input and more and more information about the target. The way information flows through the successive convolutional blocks acts as an information distillation pipeline, where the successive maps are transformed, so that irrelevant data is filtered out.

DisplasiaNet is conformed of four convolutional blocks, each with 16 channels. By plotting the 192 different feature maps of the model for each image in the interpretability dataset, a pattern for filters that activate can be established for Dysplastic and Normal cells. From this pattern, we can extrapolate what the filter is detecting, thus proving that the model focuses on relevant features contained in the image.

3.4. Low dimensionality projection of the deep features

As the successive convolutional blocks of the model filter out the irrelevant data, the output of the last pooling contains all the necessary features to perform the classification. In the typical application of CNNs, this output would be fed into a series of fully connected layers to make predictions. However, visualizing the features learned by the fully connected layer is highly complex as the relation between features is performed in a latent space.

The technique t-distributed Stochastic Neighbour Embedding (t-SNE) [57] allows us to perform a dimensionality reduction and project the learned interpretation of the different outputs of the convolution. In simplified terms, by applying this method, we are able to project onto a lower dimension, 2D or 3D, the output of the convolution for each image in the interpretability database and search for similarities. The output of the last polling layer contains 4,906 elements ($A_i = (16{,}16{,}16)$).

T-SNE is an iterative nonlinear reduction that preserves local similarities within the input data. Firstly the algorithm measures the pairwise distance, initially Euclidean distance, between different features and converts it into a probability distribution. Similar features are assigned higher probabilities, while dissimilar features are assigned a lower probability. Secondly, by minimizing the Kullback-Leibler divergence, which measures how one probability distribution is different from a second, a similar probability distribution is created on the defined low-dimensional map, the t-SNE map.

It is important to notice that t-SNE is an explanatory data method used to understand high-dimensional data. Since t-SNE is non-parametric, there isn't a function that maps data from an input space to the map, meaning it can not be directly used to make predictions. However, in L.van der Maaten [58], an approach is taken by training multivariate regression to predict the mapped location from the input data.

We can obtain a plotted distribution of the 56 images by applying this method to the interpretability dataset. The location where the image's internal features are plotted indicates a high feature resemblance with the nearest neighbors. Meaning similar images are plotted close by.

3.5. Saliency Maps

Saliency maps are a useful and popular visualization tool for gaining insights into how a particular neural network model has made an individual decision. As a noun, Salience is defined as the quality of being particularly noticeable or important by the Oxford language dictionary.

In deep learning, saliency refers to the unique features (pixels, resolution, etc.) of an image in the context of visual processing. These unique features depict the visually alluring locations in an image. Therefore, a saliency map is a topographical representation of those noticeable features and are usually rendered as heat maps where hotness corresponds to regions that have a larger impact on the model's final decision.

Saliency maps as a tool for deep learning interpretability were first introduced in [39]. This paper managed to show a method to initialize GrabCut-based object segmentation without the need to train dedicated segmentation or detection models by means of clustering the information retrieved from an image-specific saliency map.

Theoretically, these kinds of maps are obtained by backpropagating the gradient of the class score with respect to the input image pixels (Eq. 3). This is possible since CNNs preserve spatial information, the relative position of pixels to each other. The result obtained is, to an extent, an activation map in which each pixel's value describes to a degree the contribution to the classification of the input image.

$$E_{grad}(I_o) = \left.\frac{\delta S_c}{\delta I}\right|_{I=I_0} ; \qquad where \begin{array}{l} I \rightarrow Image \\ c \rightarrow Class \\ S_c(I) \rightarrow Score \end{array} \qquad (Eq. 3)$$

However, this initial method, Vanilla Saliency, poses a saturation issue when the CNN is trained with the activation function ReLU [59]. Since this function is non-linear, any neuron with a negative value gets discarded. Given that two neurons' weight is -1 and their bias is 1, the ReLU activation function will deem the output to be 0. Therefore, the gradient of these specific gradients will also be 0, and the Saliency method would indicate that this neuron is unimportant.

An approach to improve the detection of this method is to use the Smoothgrad method [40]. This method aims to make the gradient-based explanations less noisy by creating multiple versions of the original image and adding artificial noise to each of them. The pixel attribution maps of each of the images are then computed and averaged. The output is a smoother saliency map with a larger coverage area that filters out gradient fluctuation. This effect can be observed in Figure 11.

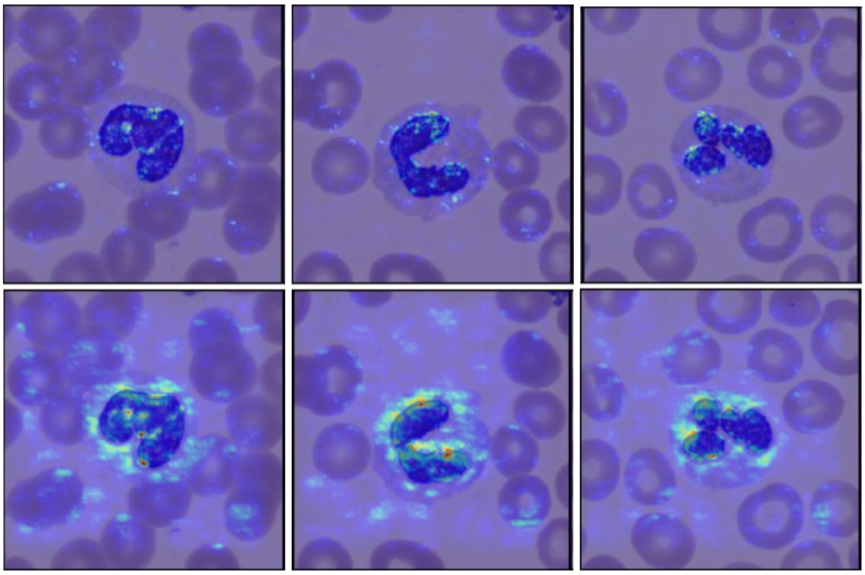


Figure 11. Comparison Between Vanilla saliency maps (top) and Smoothgrad Saliency Maps (Bottom)

This method is described by (Eq. 4), where different noise vectors (Eq. 5) are sampled from the Gaussian distribution. The parameters to tune this method are the noise level (σ) and the number of samples (N). The authors suggest that the relation between the noise level and the pixel distribution should be between 0.1 and 0.2 followings (Eq. 6). The number of samples should also not be higher than 50 since the performance tanks above the value.

$$R_{sg}(x) = \frac{1}{N}\sum_{i=1}^{n} R(x + gi)\,; \;\; where \;\; \begin{array}{l} N \rightarrow \; samples \\ x \rightarrow image \\ gi \rightarrow Noise\ sample \end{array} \qquad (Eq.\ 4)$$

$$gi \sim N(0, \sigma^2); \qquad where \;\; \begin{array}{l} N \rightarrow \; samples \\ \sigma \rightarrow \; noise\ level \end{array} \qquad (Eq.\ 5)$$

$$0.1 \leq \frac{\sigma}{x_{max} - x_{min}} \leq 0.2; \quad where \;\; \begin{array}{l} x_{max} \rightarrow \max pixel\ value \\ x_{min} \rightarrow \min pixel\ value \end{array} \qquad (Eq.\ 6)$$

After the testing, different values for image samples (N) and noise level(σ) values that return the best saliency maps are N=40 and σ=0.15. The application of this technique aims to highlight different structures between the Dysplastic cells and the Normal cells.

3.6. Occlusion techniques

Occlusion sensitivity techniques enable the creation of relevant region maps in a model agnostic matter. The principle relies on iterating a certain image, usually out of the training or test set, repeatedly through the model, just varying a slightly occluded region in the input (Figure 12). The idea behind these techniques is to manifest the importance of some regions of the image and their contribution to the overall model prediction. The score of each iteration is recorded onto the output image in the exact location as the occluded region in the input, creating a heat map. Relevant regions in the image will have a more considerable impact on the prediction. Thus, the completed output image of the iterative process will highlight the sensitive areas of the input image.

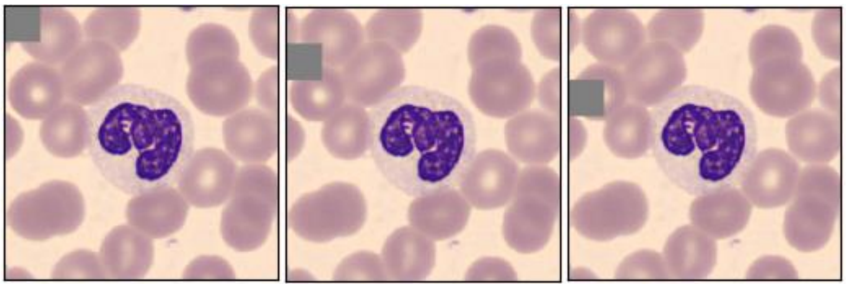


Figure 12. Occlusion Sensitivity Maps patch iteration

There are many techniques and parameters that can be altered to achieve these maps. However, it is crucial to bear in mind that some occlusion methods can trick the model into finding relevant structures within the occluded region, therefore invalidating the results of the obtained map. This effect can be significant in convolutional architectures. Different low-level filters might have specialized in detecting certain patterns, such as straight lines, which grant specific neurons with a higher weight.

3.6.1. Occlusion Sensitivity Maps (OSM)

One of the first documented techniques, occlusion sensitivity maps (OCS) [32], shifts a grey squared patch over the entire input image. For black and white images, the grey color is obtained by splitting the pixel range in half. For RGB images, the 'grey' value is obtained following the same process but performed individually for each channel. This technique leaves a unique parameter to be tunned with, the patch size. A balance must be found within what the size of a relevant feature is, and how much of the image we are going to occlude. Another aspect to bear in mind is the tradeoff between the patch size and the computation time. For our particular case, the number of iterations increases at a rate of $N_{iterations} = 89401/S_{patch}^2$ (eq.5), with small patches being very computationally demanding. The results of performing this technique can be observed in Figure 13. Notice that this method usually normalizes the output result; therefore, the representation lacks any numerical values. However, we have maintained the results un-normalized to visualize the impact on the score for each case.

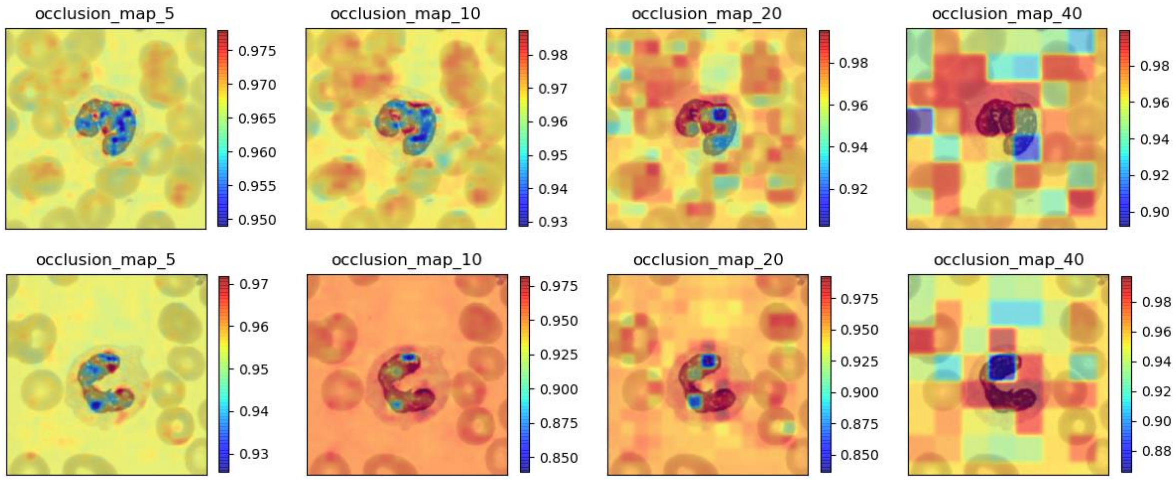


Figure 13. OSM grey patch 5px, 10px, 20px, 40px [TD_BNE_2249981.jpg (Top), TD_BNE_2256328.jpg (Bottom)]

Figure 13 presents OSM with gray patches for two dysplastic cells. The selected patch sizes are 5px^2, 10px^2, 20 px^2, 40px^2. As can be observed, small patches return smoother heat maps. However, the utilization of gray patches returns undesired results. The benchmarked score for the top and bottom images without occlusion are 0.966 and 0.954, respectively, for the dysplastic class. It is expected to obtain no difference score difference for the regions where the neutrophil is not present. Instead, when the image's background is occluded, an improvement of up to 4% is recorded for the class score. Nonetheless, a decrease in the score is recorded when the patch is located over the nucleus of the neutrophil, indicating it is a relevant feature for the classification of cells.

As explained previously, this method has a significant drawback, mainly induced by the convolutional sector of the model. The color gray is a color that is not typically found in the dataset. Introducing it in a patch can return unexpected results, as evidenced by the previous sensitivity maps. At this point, the color of the patch can be tuned. There are many ways this can be done, Zhong *et al.* [60] attenuated the original images with random noise for data augmentation and training. As granularity is an important feature, as stated by the pathologic experts, it is not a viable method as it can induce erroneous predictions. Yun *et al.* [61] substitute the grey patch with a patch located in the same position but from another image. This is not an ideal solution as our dataset is relatively constant regarding the disposition of the image; the image is centered with the cell, and the nucleus is usually found in the center of the image. Therefore, the patch substitution would not modify the image enough. Another option that complies with our objective is to substitute the patch with the mean value of the patch in the original image. This will remove any characteristic texture without introducing new color elements into the image.

3.6.2. Local Interpretable model-agnostic explanations(LIME)

A more state-of-the-art occlusion technique is LIME (Local interpretable model-agnostic explanations) [41]. The basic principle of this method is the same as for OSM, where regions are occluded, and the output score is evaluated. A quick image segmentation algorithm generates superpixels that will act as the occlusion patches (Figure 14). When choosing the segmentation algorithm, Quickshift [52] in our case, we aim to maintain the overall geometry of the image without introducing sharp edges into the image. While configuring the segmentation algorithm, we set the different parameters to achieve relatively small areas that cluster pixels in the same color neighborhood. The output is constituted by more 'natural' regions that can be found on the cell morphology. These are then replaced by the average color in the superpixel, thus addressing a previous issue.

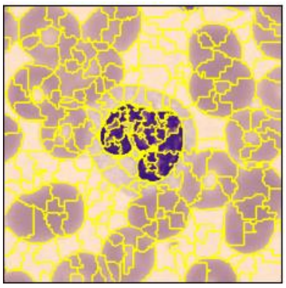


Figure 14. Quickshift superpixel generation (TD_BNE_2249981.jpg)

This method also introduces a novel way of iterating the image. Several superpixels can be deactivated simultaneously, unlike the previous method, where only one patch is present in each iteration. We can then configure the probability of a pixel being deactivated and the number of iterations to process the input image. The average number of superpixels in each image after the quick-shift algorithm is 227. Seven hundred iterations of each image are computed where randomly different patched are activated and deactivated. The score for each iteration is then plotted onto a 2d space and fitted with a more straightforward linear model. From this model, we can extract the top superpixels, which, when occluded, affect the most the output result. The features contained in that superpixel are the ones the model focuses on the most when making predictions

3.7. Class Activation Maps (CAM)

Class activation maps (CAM) is another approach to visualize what features a DL model uses to perform a specific prediction. In general terms, it is a pixel heat map composed of the different feature maps extracted from the last convolutional layer filters. Different filters react to different image features. Combining the highest weighted feature maps can give us a sense of what the model reacts to for certain classes.

The original method [33] required the model to be modified by replacing the fully connected layers at the end of the model and replacing them with a single Global Average Pooling layer containing the different class outputs. This substitution aims to have a direct relationship between the last layer convolutional filters and the output classes via the weight each filter is attributed for each class. However, this method has a significant drawback and is that the model is required to be re-trained. As the model is being retrained with a simpler architecture, many hidden feature relations are eliminated. Also, from the interpretability analysis point of view modifying the model defeats the purpose.

New methods have been developed based on this original method that do not require a model re-training. Some of these are Grad-CAM [34] and Score-CAM [36].

3.7.1. Grad-CAM

Grad-CAM [34] is based on the same principle as CAM to obtain the class-discriminative localization map. However, this new methodology (Figure 15) computes the class score's gradient (Eq. 7) with respect to the obtained feature maps from the convolutional layers. The different gradients are then globally pooled across the width and height dimension to obtain the weights of each of the filter channels (Eq. 8). Like CAM, the output map is a weighted combination of the feature maps combined with ReLU (Eq. 9). Finally, the resulting heat map is upscaled to match the Input image dimension. This novel approach generalizes the CAM method and avoids having to modify and retrain the model.

$$\frac{\partial y^c}{\partial A^k};\ where\ \begin{matrix} A^k \rightarrow Feature\ Map\ Activation \\ y^c \rightarrow class\ score \end{matrix} \qquad (Eq.\ 7)$$

$$w_k^c = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}\ ;\ where\ \ w \rightarrow Feature\ Weights \qquad (Eq.\ 8)$$

$$L_{GradCAM}^c = ReLU\left(\sum_k w_k^c\, A^k\right);\ where\ \ k \rightarrow Feature\ Maps \qquad (Eq.\ 9)$$

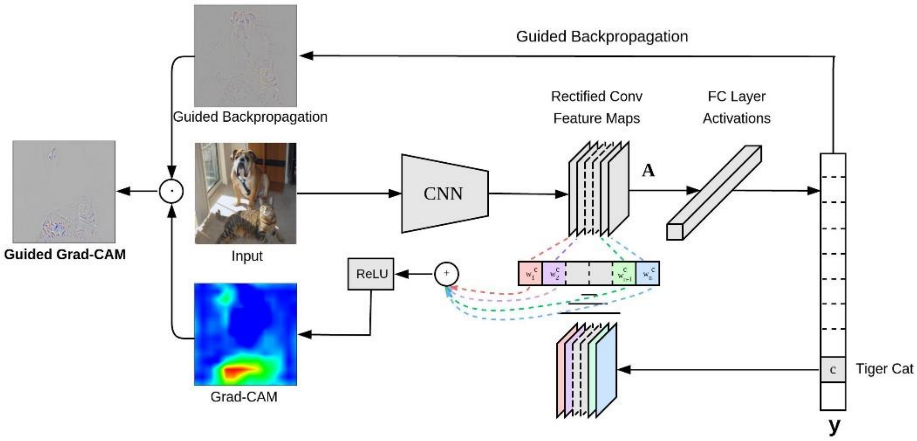


Figure 15. Grad-CAM methodology scheme (Source: [23])

The literature states that this method lacks stability, as when several occurrences of a class are present, its performance drops, and for single object images, the method fails to highlight the whole region on the heat map.

3.7.2. Grad-CAM ++

Grad-CAM++ [35] remedies the flaws mentioned above of its predecessor by reformulating (Eq. 8) by adding a pixel-wise weighting distribution to on the final convolutional feature map (Eq. 10). The weighting coefficients of the pixel-wise gradient only use positive gradients. By doing so, the neuron's output is increased rather than suppressed, capturing the most important visual features.

$$w_k^c = \sum_i \sum_j \alpha_{ij}^{kc}.ReLU\left(\frac{\partial y^c}{\partial A_{ij}^k}\right) \quad where\ \alpha_{ij}^{kc} \rightarrow pixelwise\ weighting\ coef \qquad \text{(Eq. 10)}$$

This generalized method of Grad-CAM is able to provide better visual explanations of CNN model predictions in terms of better object localization as well as explaining occurrences of multiple object instances in a single image,

3.7.3. Score-CAM

ScoreCAM [36] offers a different approach to the matter in how the original CAM algorithm is evaluated. This novel method achieves a better visual performance with less noise and offers better stability than Grad-Cam and Grad-CAM++. This is achieved by replacing the gradient-based weight with a score-based weight that has a much smoother curve of change.

The basic workflow of Score-CAM can be observed in Figure 16. This method first obtains the different feature maps of the last convolutional layer by forward propagating the image through the model (Phase 1) After obtaining the different feature maps, these are up sampled from 16x16px to the original 299x299px dimension of the input image by performing a bilinear-interpolation. The resultant activations maps are normalized within the range [0,1] by employing (Eq. 11).

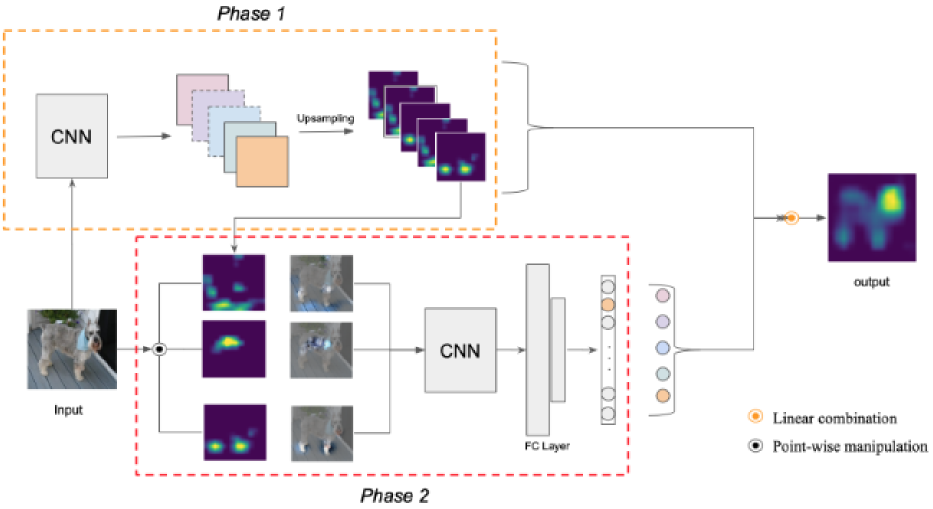


Figure 16. Score-CAM methodology scheme (source:[36])

$$A_{i,j}^{k} = \frac{A_{i,j}^{k}}{\max A^{k} - \min A^{k}} \quad where\ A^{k} \rightarrow feature\ map \qquad (Eq.\ 11)$$

Secondly, following an iterative process, the method masks each feature map over the original input image by performing an element-wise multiplication between each of the upsampled and normalized feature maps and the input image. The different obtained masked images processed through the original CNN with a SoftMax output(Eq. 12), annotating the obtained score result for each feature map (Phase 2). The final map obtained is a weighted addition of the different feature maps, where the weights (Eq. 13) are given by obtained score of phase 2. The final images are processed with a ReLU activation function to eliminate any negative values (Eq. 14). This method is only interested in returning features that have a positive influence on the class of interest. A higher difference in the output score implies a larger importance of the masked region.

$$S_{k} = softmax\left(F\left(M^{k}\right)\right) \quad where\ M^{k} \rightarrow Upsampled\ Masked\ Image \qquad (Eq.\ 12)$$

$$|W_{k}^{c} = S_{k}^{c} \quad where\ w_{k}^{c} \rightarrow Weight\ of\ the\ featurew\ map \qquad (Eq.\ 13)$$

$$L_{scorecam}^{c} = ReLU\left(\sum_{k} w_{k}^{c} A^{k}\right) \rightarrow Weight\ of\ the\ featurew\ map \qquad (Eq.\ 14)$$

3.8. Background generalization test

The final test performed on DisplaciaNet is a background generalization test. With this test, we aim to prove that the background does not influence the scoring prediction of the model, meaning the NN is able to generalize the background artifacts. The rationale behind this method is that if the model can generalize the background, there should be no score difference between any image used to train the dataset and the same image that has had the background removed.

For the test, the 56 images of the Interpretability dataset have been manually segmented to remove any artifact present in the background, obtaining an image that only contains the object of inspection, the Neutrophil, as shown in Figure 17. The artifacts located in the background, for the most part, are red blood cells, although platelets and other types of white blood cells. These new images contain the Neutrophil in the center of the image and a uniform color for the background that matches the base color of the original image's background.

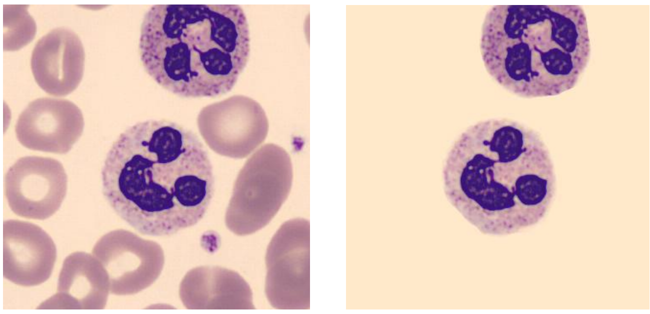


Figure 17.Two Neutrophils in the same Image, with and without background (ZN_SNE_4223913.jpg)

4. Software Implementation

This chapter overviews the Cell Annotation App design, instructions on how to be used, and instructions on how to install locally. Cell Annotation App GitHub repository [60], Cell Annotation App Docker Image [61]

This chapter also overviews the general structure of the Interpretability Study Github repository [62]

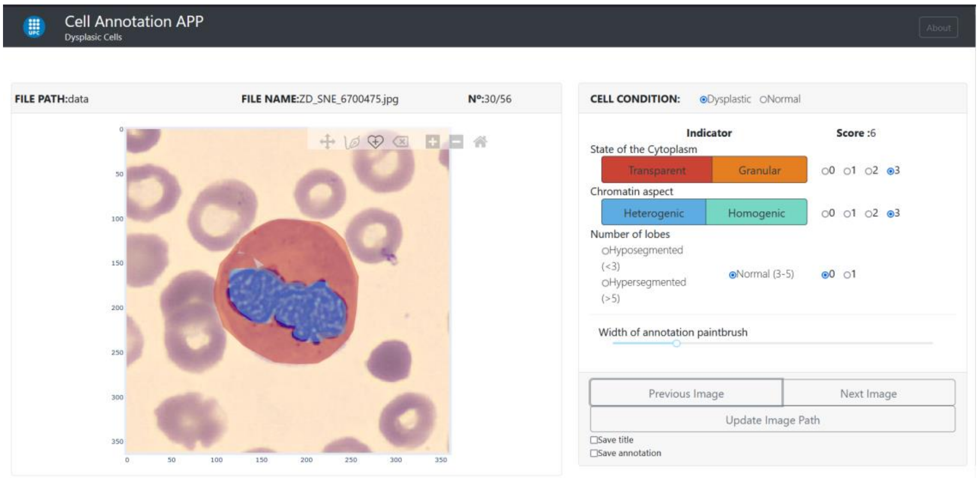
4.1. Cell Annotation App

An annotation application has been developed to compare the decision criteria of the DisplasiaNet NN with the expert pathologist decision criteria. This annotation application, *Cell Annotation APP*, is built using Dash [50], a productive web-based python framework, and packed and deployed onto an online server located at the EEBE campus of UPC for online access through the Docker platform [51].

The application aims to collect the information the pathologist considers t the most relevant o diagnose a neutrophil cell as dysplastic or normal. In section 1.4. the most relevant features between a healthy and a dysplastic neutrophil have been discussed. In unison with the pathologists at the Hospital Clinic of Barcelona, the three main features do discern whether a cell presents dysplasia or not have been selected. The following are listed in order of priority, presence/absence of granules in the cytoplasm, how densely packed the chromatin is in the nucleus, and the number of lobes the nucleus presents. Each attribute is also to be scored by the user depending on how predominant each attribute presents itself. Regarding the granularity and the chromatin state, the score ranges from zero to three, zero being not present and three being very present. For the number of lobes, a score of zero is given if the nucleus presents a standard number of lobes, between three and five, and a score of one is given for an abnormal number, hypolobulated(<3) or hyperlobulated(>5).

4.1.1. GUI Application Description

The GUI, as seen in Figure 18. Cell Annotation APP GUI, is divided into three main parts, a header(top), a visualization card (left) and, a tools card (right). The header contains the applications' title and the UPC logo, which acts as a hyperlink to the project web page. It also includes an *About* button that opens a pop-up window containing detailed instructions on how to use the application and valuable tips. (Annex A)



Cell Annotation APP
Dysplasic Cells

FILE PATH:data FILE NAME:ZD_SNE_6700475.jpg Nº:30/56

CELL CONDITION: ●Dysplastic ○Normal

Indicator Score :6

State of the Cytoplasm: Transparent Granular ○0 ○1 ○2 ●3

Chromatin aspect: Heterogenic Homogenic ○0 ○1 ○2 ●3

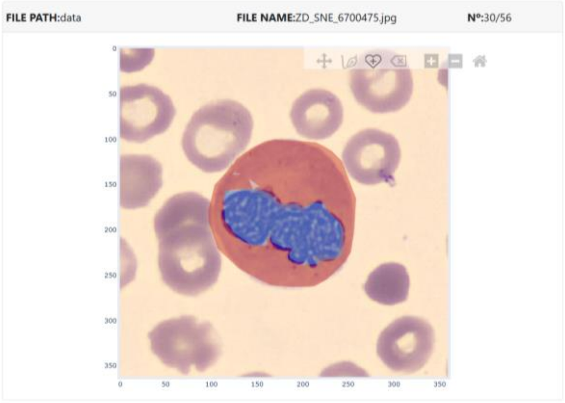
Number of lobes: ○Hyposegmented (<3) ○Hypersegmented (>5) ●Normal (3-5) ●0 ○1

Width of annotation paintbrush

□Save title
□Save annotation

Figure 18. Cell Annotation APP GUI

The Visualisation card allows the user to visualize the cell images and interact with them. On the upper region of the card, relevant information of the image can be observed. These include the path where the image is stored (*File Path*), the image reference name and extension(*File Name*), as well as the image counter (*Nº*), which indicates the number of images in the file directory and the current position of the image that is being visualized. The central part of the card contains the image display and fulfills two purposes. The first is to show the selected cell image, and the second is to enable the user to manually draw annotations over the image. A toolbar hoovering above the image display allows the user to configure the image visualization (zoom in, zoom out, pan, reset view) and customize the annotation tool (free draw tool, closed-loop tool, and erase selected annotation).

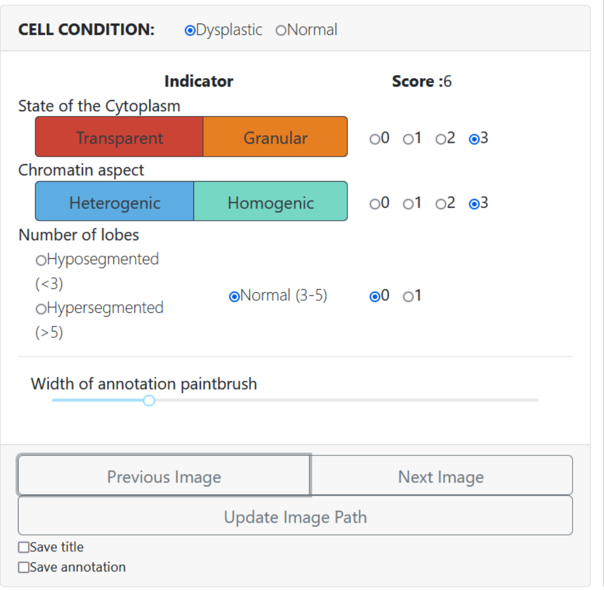


FILE PATH:data FILE NAME:ZD_SNE_6700475.jpg Nº:30/56

Figure 19. Cell Annotation APP GUI - Display Card

Lastly, the tools card enables the user to document the process of diagnosing dysplastic cells thoroughly. The top half of the card is dedicated to the cell diagnostic. The user can specify the cell condition (*Dysplastic/Normal*) as well as select the color to annotate the image. Each symptom has had a specific color assigned to it. Orange is used to annotate the absence of granules in the cytoplasm and red for their presence. Blue is used to annotate a loosely packed chromatin in the nucleus, displayed in the image as heterogeneous staining. On the other hand, green is used to annotate a homogeneous stained nucleus.

The bottom half of the tools card contains a set of buttons that enable the user to navigate the image directory. It also includes a slider to adjust the width of the annotation paintbrush. The checkboxes, *Save Title* and *Save Annotation*, save the original image with the inputted information and the drawn annotation overlaid respectively as a jpg image.



CELL CONDITION: ● Dysplastic ○ Normal

Indicator **Score :6**

State of the Cytoplasm: ○0 ○1 ○2 ●3

Chromatin aspect: ○0 ○1 ○2 ●3

Number of lobes: ○ Hyposegmented (<3) ○ Hypersegmented (>5) ● Normal (3-5) ●0 ○1

Width of annotation paintbrush

□ Save title
□ Save annotation

Figure 20. Cell Annotation APP GUI - Tools Card

The application is hosted on the DeepBox Server at the EEBE faculty as a docker container [51]. In order to access it, the user must first connect to the UPC's VPN (Virtual Private Network) UPCLink and access the server's port 8050 from the browser [53]. The images used for this study have been preloaded on the server and are accessible from the dash application. The annotated information is stored in the exact path as the images are found and can be downloaded from the server using any VNC file transfer system.

4.1.2. Annotation procedure

To annotate, the user must follow a simple workflow (Figure 21). This workflow has been designed to standardize the process and obtain consistent annotations between the different pathologist's annotations. The GUI has been designed to maximize this workflow and reduce error or confusion.

First, the user must click on update images; this will refresh the images in the server directory and load the first image on the display frame. The user must then inspect the cytoplasm to decide whether it presents granules (*granulated*) or lacks them (*transparent*). Once the cytoplasm state option is selected on the tools card, the user can outline the cytoplasm with the mouse. When the user has finished outlining the cytoplasm, it will appear red if transparent was selected or orange if granulated was selected. Finally, the user must grant a score to the selected feature, zero being not predominant and three being very well represented.

The user must then inspect the nucleus. First, the user must observe the nucleus and decide whether the chromatin has been stained in a homogenous matter o a heterogenic matter. Posteriorly they must outline the nucleus, the result will be a green nucleus for a homogenous stain and blue for a heterogenic stain. A score must be granted to the feature, zero being not predominant and three being very well represented. Finally, the lobes in the nucleus must be counted. The cell presents hypolobulation if there are less than three lobes(<3), and hyperlobulation if it presents more than five(>5). If it presents between three and five, it will be classified as normal.

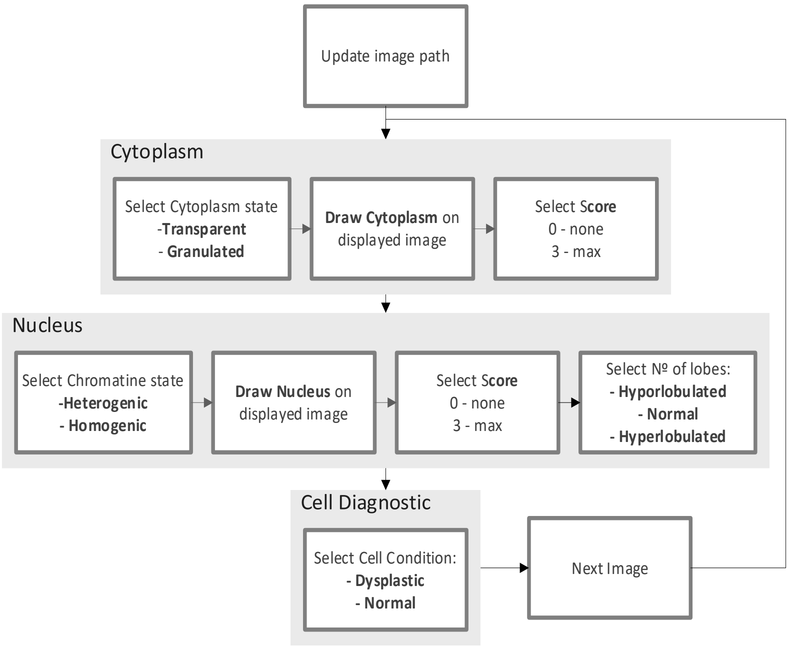

Update image path
Cytoplasm
Select Cytoplasm state -Transparent - Granulated
Draw Cytoplasm on displayed image
Select Score 0 - none 3 - max
Nucleus
Select Chromatine state -Heterogenic - Homogenic
Draw Nucleus on displayed image
Select Score 0 - none 3 - max
Select Nº of lobes: - Hyporlobulated - Normal - Hyperlobulated
Cell Diagnostic
Select Cell Condition: - Dysplastic - Normal
Next Image

Figure 21. User Annotation Workflow

To finish the cellular exploration, the user must then diagnose the cell as Dysplastic or Normal with the help of the score information. To continue the next cell image exploration, the user can press the *Next Image* button located on the tools tab.

The information introduced into the app during the exploration is automatically stored as a *.json file on the server. The file is structured as a python dictionary, where the different entries are the different pieces of information inputted through the GUI. The annotations made on the image are stored as a vectorized image.

Further modifications to this application are planned to be introduced to the GitHub repository [54]. These include an upload button, where the user can select a compressed file containing the images to be annotated, and a download button that downloads a compressed file containing the different annotations. A segmentation algorithm to automatically annotate the nucleus and the cytoplasm would significantly increase the annotation speed as the pathologist would only be required to press buttons and not manually have to draw on the image.

4.1.3. Docker Installation

Docker is a platform that offers service products that use OS-level virtualization to deliver software in packages called containers. These containers can be locally installed on devices or installed on a server to be accessed remotely through the internet. As dockers containers are isolated distros, the designer can install any operating system with specially developed applications and all the necessary libraries without interfering with the host device. It simply offers an easy way of sharing complex applications.

Cell Annotation App has been packed into a docker container and can be downloaded from the online repository [55] to be run locally on any device. The image consists of a lightweight (1.64GB) Ubuntu Linux distribution with an amd64 architecture. Python and the rest of the libraries used in the app are pre-installed.

To do so, the user must previously install the docker desktop program on the hosting device [56]. This suite allows the user to manage, start and stop the different containers downloaded onto the host computer. However, the initial setup must be done through the native OS command prompt.

First, the user must download the Cell Annotation APP container from the docker hub repository. This is done in windows with the command docker pull stevecreations/tfm_cell_annotation_app:gamma can be used. For Linux, Sudo permissions must be granted with the same instruction.

Secondly, the user must mount the container onto the host computer and configure the connexion display port and the shared volume path. As the container works wholly isolated from the host device, a connection must be established to visualize the GUI and transfer the images and annotations between OS. The cell annotation app uses port :8050 to display the application, and fetches the images, and saves the annotation files from "/usr/src/app/data”. The user must then decide on a folder path to use as the shared folder between the host and the container. The user must also decide on a connection port on the host computer to map to the container port. It is recommended to use port :8050, although the user must ensure that no other application is using this port through the command prompt instruction netstat -oan, on both windows and linux. Using the instruction docker run -dit -p “host_computer_port”:8050 -v "host_computer_path":/usr/src/app/data stevecreations/tfm_cell_annotation_app:gamma. The container is mounted and ready to be used.

Finally, the user can access the application through any web browser typing in the address localhost:”host_computer_port” or 127.0.0.1:”host_computer_port”.

From the docker desktop suite, the application can also be accessed, stopped, and re-ran. (Figure 22)

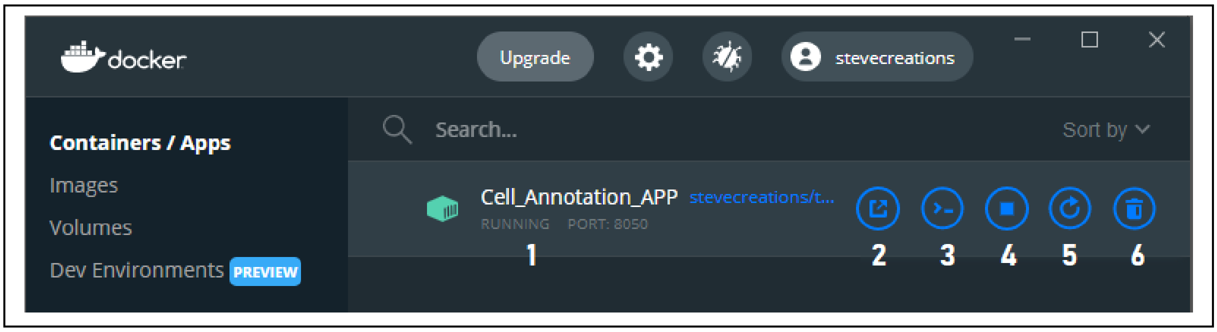


Figure 22. Docker desktop GUI. 1- Name of the Docker Container and Status, 2- Open Container on browser, 3- Open CLI, 4- Stop container, 5- Restart Container, 6- Delete Container from computer

4.2. Interpretability Study

The different interpretability methods applied as well as the different analyses performed, have all been developed using Python 3.8.8 programming language. The different scripts specially developed for this interpretability study have been pushed to the DysplaciaNet_Interpretability_Study GitHub repository [62].

The Github repository contains 2 main scripts:

- ***Interpretability.py***: is the main scrip of the project. This file is split into different sections, each belonging to a technique described in the methodology chapter
  - *Section 1.* initializes the script.
  - *Section 2*. loads and compiles the DysplaciaNet model. A Boolean value can be modified to visualize the summary of the model and the needed FLOPs to perform one prediction.

- *Section 3*. Contains the definition of the function that allows to make a prediction on a single image.
- *Section 4*. Contains the definition of the function that allows to visualize the different Feature maps for each convolutional layer.
- *Section 5.* Contains the definition of the function that allows to perform the Occlusion Sensitivity Maps.
- *Section 6.* Contains the definition of the function that allows us to create the different Saliency Maps . (S.6.1 -Vanilla Saliency, S.6.2- Smoothgrad saliency)
- *Section 7.* Contains the definition of the function that allows to perform different Grad-CAM maps. (S.7.1 -Grad-CAM, S.7.2- Grad-CAM++)
- *Section 8*. Contains the definition of the function that allows us to create Score-CAM Maps
- *Section 9*. Contains the definition of the function that allows us to create LIME Maps.
- *Section 10*. Contains the definition of functions that allow us to process the *.json file created by the Cell Annotation APP. From this file, we can extract the different scores for the relevant features as well as the masks for the cytoplasm and nucleus segmentation.
- *Section 11*. Contains the main loop. This section of the code is where all the different analysis are called from. The Loop first loads an image from the path given by the input array and then performs all the different analysis on the image. By using the plot_result.py an image as well as a numpy array file are stored in the same path as the image containing the results of each analysis.

- ***Display Analysis.py*****:** contains the different functions to mask and plot all the different analysis for the images in the Interpretability database. The output of this function has been included in the different annexes of this document.

The different libraries used in the project are:

- *Tensoflow.keras*: is the designated high-level Tensorflow API. This library is used to interact with DisplaciaNet. Licensed under the Apache License V2.0
- *tf_keras_vis*: is a GitHub repository community contributed to visualize different parameters of CNN. For this library, we have used the functions to perform the Grad-CAM, Grad-CAM++, and Score-CAM analysis of DisplaciaNET. Licensed under the MIT license.

- *Lime*: is a GitHub repository that contains the functions to perform the LIME analysis to DisplaciaNet. Licensed under the BSD 2-Clause "Simplified" License. Copyright (c) 2016, Marco Tulio Correia Ribeiro
- *skimage*: is a collection of algorithms wildly used for image processing. From this specific library, we used the functions to perform the T-SNEdimensionality reduction of the convolutional output. Licensed under the BSD 3-Clause "New" or "Revised" License. Copyright (C) 2019, the scikit-image team.
- *Matplotlib*: is a comprehensive library for creating static, animated, and interactive visualizations in Python. This library is used to plot all of the images derived from the analysis of DisplaciaNet contained in this document. licensed under the BSD compatible code License
- *Numpy*: is a library specifically developed to perform scientific computing in python. Licensed under the BSD compatible code License

5. Results

This chapter contains the results of applying the algorithms and methods described in chapter 3. Methodology. For each method, the result is discussed individually.

5.1. Cell Annotation App results

These 56 images have been reannotated by the Expert Pathologist at the Hospital Clinic de Barcelona using the Cell Annotation APP. Each image has been attributed a total score, resulting from the addition of the cytoplasm granularity score, nucleic chromatin score, and the lobe segmentation score. The total score is evaluated on a uni-dimensional scale ranging from -6, normal cell, to 7, a very dysplastic cell.

It is important to note that this scoring scale has not been based on any existing medical scoring system used in Haematological diagnostic procedures. This system has merely been designed as a method to describe the most significant features used by pathologists, with the aim of having a point of reference to compare features against DisplasiaNet.

The summary of the annotated scores can be observed in Table 3. The complete annotation table has been attached in Annex B. As can be seen, the total score value returns a good classification. This scoring system has managed to cluster the 56 images based on the addition of individual feature scoring. There is no overlap between the different clusters, Dysplastic Cells, and Normal cells, as the lowest score for dysplastic is 3 and the highest score for normal is zero.

Out of three individual feature scorings, Granularity in the cytoplasm is the best performing as expected. However, this feature does not fully manage to cluster the different cell classes clearly, but it does a reasonably good job. A slight overlap can be appreciated as the lower scoring normal cells and the highest-scoring dysplastic cells present a small granularity (score of -1). Nonetheless, the lobular segmentation and the chromatinic density aid in deeming dysplastic cells as dysplastic, setting them apart from the normal ones.

As for lobe segmentation and chromatinic density, there is no clear line that separates both clusters. It is noted that there is a higher chance of a cell presenting lobular segmentation anomalies when the cell is dysplastic. The chromatin density also appears to be much clumpier and loose on a cell that presents dysplasia.

	Class	Mean	Min.	Max.	Stdev.
Total Score (-6 to 7)	*Dysplastic*	5.4	3	7	1.37
	Normal	-2.6	-5	0	1.50
Cytoplasm Score (-3 to 3)	*Dysplastic*	2.4	-1	3	1.06
	Normal	-1.9	-3	-1	0.56
Nucleus Score (-3 to 3)	*Dysplastic*	2.2	-1	3	1.07
	Normal	-1.0	-3	2	1.34

Table 3. Summary of Scores extracted with Cell Annotation APP

It is also worth noticing that three out of the ten selected misclassified images were relabeled as correct during the expert annotation process(Figure 23). For image WD_SNE_2288170.jpg, the high chromatinic density indicates the cell has entered the Apoptosis stage, programmed cell death in multicellular organisms. Image WD_SNE_2426212.jpg presents high granularity and a normal lobe segmentation. Image WN_BNE_3328133.jpg, although it presents high granularity, the chromatinic density and the lobular segmentation are key indicators of dysplasia.

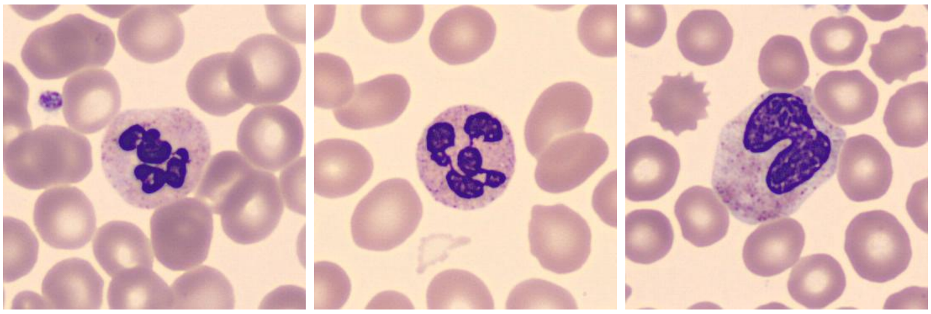


Figure 23. Images relabeled by the Expert pathologist. Left:Relabeled Normal(WD_SNE_2288170.jpg) Center: Relabeled Normal(WD_SNE_2426212.jpg) Right: relabeled dysplastic(WN_BNE_3328133.jpg)

5.2. DisplasiaNet Score vs Cell Annotation APP score

The scoring scale designed for the Cell Annotation APP has proven to cluster the dysplastic and normal cells clearly. As is, the first logical approach towards demonstrating that DisplasiaNet focuses on the same features as the pathologist when diagnosing dysplasia is to compare both scoring scales directly.

Employing a scatter plot, we can project on each axis the score result for DisplasiaNet (X-axis) and the Cell Annotation APP (Y-axis). If there is a direct correlation, the different points should more or less align on the plot area. However, this is not the case.

Figure 24 contains the scatter plots for the Total Score vs the DisplasiaNet score; the Cytoplasm Granularity Score vs the DisplasiaNet score; and the nuclear chromatinic density score vs the DisplasiaNet score. The different classes in the plot correspond to Dysplastic cells(orange), Misclassified Dysplastic cells (Green), Normal cells (Maroon), and Misclassified Normal cells(yellow). Values above Zero for the pathologist score axis correspond to Dysplastic cells, and values below correspond to Normal cells.

We can observe that the top left corner contains a cluster of Dysplastic cells. However, the distribution on the chart does not indicate a correlation between the scoring systems used in the cell annotation app and the DisplasiaNet score. For it to correlate, the pathologist score should decrease as the DisplasiaNet score increases. The same inverse behavior should apply to the cluster of normal cells located at the bottom right corner of the graphs. As for the misclassified cell images, the different clusters are located where expected, top right for misclassified dysplastic cells and bottom left for misclassified normal cells.

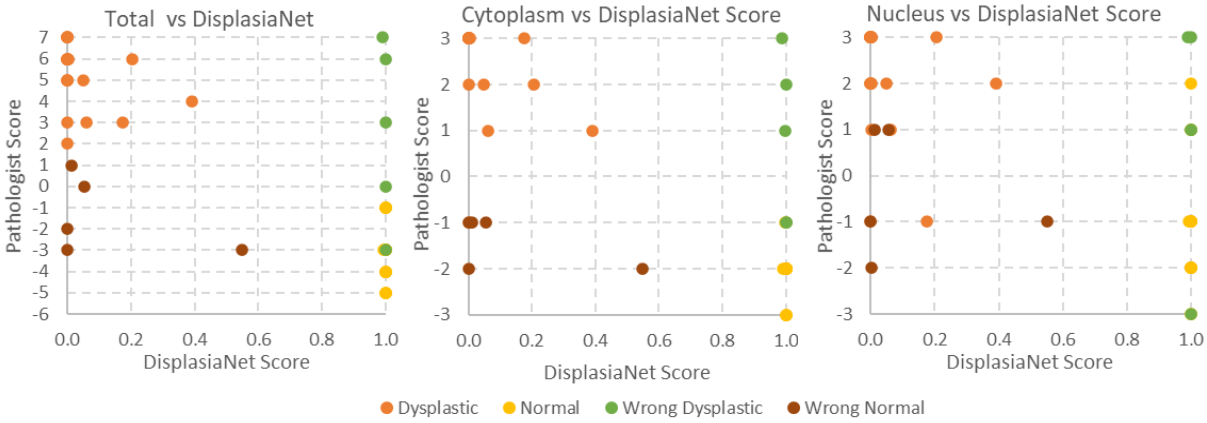


Figure 24. Graph comparing the score from the annotation applications VS the Score obtained from DisplasiaNet. Total Score(left), Cytoplasm Score(Center), Nucleus Score(right)

This direct correlation method has not furnished any insightful data regarding the classification criteria of the NN. However, the DisplasiaNet score is the final distilled classification of both the feature extraction process, performed by the convolutional section and the models, and the feature classification, performed by the dense layers.

5.3. Low dimensionality projection of the Deep features

The score classification by the expert pathologist can also be compared to the output of the convolutional block. This comparison can be obtained by representing the output of the convolutional block for the different images in the dataset onto a scatter plot through a two-dimensional reduction of the 4,096 internal features employing the t-SNE algorithm [57]

Figure 25 contains the t-SNE maps for the output last convolutional block of DisplasiaNet. The different colors indicate the different image classes studied in the dataset: Dysplastic cells (Red), Normal cells (Green), misclassified Dysplastic cells (blue), and misclassified Normal cells (cyan). The values overlaid correspond to the score obtained by means of the Cell Annotation APP. The values that have been circled in purple are the wrongly classified images that the pathologist has relabeled. As can be observed, there are two major clusters, corresponding to normal cells (upper left corner) and Dysplastic cells (center). Most of the misclassified images have been placed close to their respective clusters as expected. Interestingly enough, 5 out of the 7 misclassified images are located on the lower range of the y axis, indicating feature proximity.

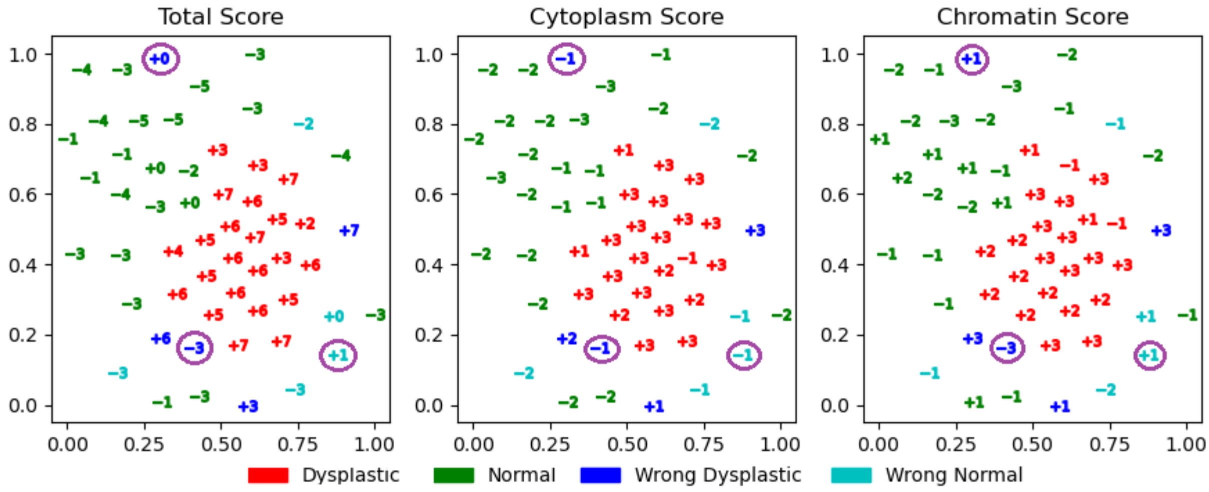


Figure 25. t-SNE map combined with the 4 class images and overlaid with the nominal score of the image given by the expert pathologist.

As for the score correlation for the different annotated features, there is no clear pattern. An additional factor contributing to this result is the fact that pathologist evaluation criteria for granularity and chromatin density is highly subjective. There is not an exact metric to evaluate these features, which introduces great variability throughout the dataset. Nevertheless, the CNN is able to isolate enough features for the model to predict, with high accuracy, the different classes.

5.4. Feature Maps

In order to obtain a feature map, the original NN must be slightly modified to output information in a matter that resembles an image [28]. The first 12 Keras layers of the model are extracted and compiled to build an independent model. These layers correspond to the four convolutional blocks conformed by the convolutional, activation, and max-pooling layers. Having a total of 16 channels per layer involves obtaining a total of 192 2d feature maps. Each feature map has been normalized and represented in the Viridis color scheme to make the visualization more palatable. This color scheme assigns yellow for high values and dark blue for low values.

Analyzing these maps will give us an approximate idea of what the NN has learned to detect as a relevant feature to classify a neutrophil with dysplasia or not. The selected 56 images have been visually analyzed and compared.

The first layer arguably retains most of the information contained in the original image. Interestingly enough, the filters highlight the different regions of interest in the image in accordance with the neutrophils' main parts, the nucleus, the cytoplasm, and the background erythrocytes. The first activation layer filters out broad information, returning images with just the nucleus, background, or even the texture of the different regions of interest. When the activation function is applied, the filtered regions become more apparent, although a blank image is returned for some of the channels. This is due, in part, to the ReLU activation function, which truncates all negative values and only returns the positive ones.
As expected, as the layers progress, more abstract features are filtered by the convolutional blocks. These advanced and more specified filters are where the differentiation between a dysplastic cell and a normal one can be observed. An example of this can be seen in the fifth channel of the third convolutional block. That particular filter searches for a granulated cytoplasm and ignores the nucleus. Figure 26 shows dysplastic cells on the left column and normal cells on the right column. Due to the agranularity presented by the dysplastic cells, the cytoplasm is hardly represented when compared to the normal cells' granulated cytoplasm.

The output of the last convolutional block is complicated to analyze as the meta-features encoded within the image are highly abstract, due in significant part to the reduction in the size of the map (23x23px) compared to the original image (299x299px).

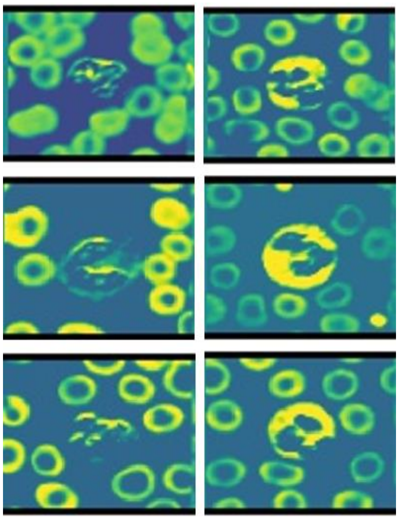


Figure 26. Difference between a Dysplastic cell and a Normal Cell - Polling layer 3- channel 5. Left (TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg) Right (TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg)

These observations, however, are merely subjective as they are solely based on a visual representation. We do not dispose of any means to know how the NN interprets the number of lobes or qualifies the granularity level of cytoplasm in the various images. Despite that, the pieces of information the different filters focus on are in accordance with the procedures the pathologists use to diagnose dysplastic neutrophils.

5.5. Saliency maps

Saliency Maps are a useful technique to highlight pixels in the input that, in the context of the NN interpretability, depict visually alluring locations for the prediction of the class image. As explained in the methodology chapter, the selected algorithm to obtain the saliency map for each image in the interpretability dataset is Smoothgrad with the parameters adjusted to : image samples N=40 and noise level σ=0.15. The complete maps can be visualized in Annex C3.

Figure 27 contains the saliency maps for dysplastic images (top) and normal images (bottom) masked onto the original image. In general terms, a great emphasis is placed upon the cytoplasm for both classes.

For Normal Neutrophils, the regions that present the highest saliency, those with the highest 'heat' correspond to areas where the granulation is high. In certain regions, the heatness of the map highlights individual highly defined granules. As for dysplastic cells, the opposite situation is highlighted; regions presenting a high saliency are those that lack granulation.

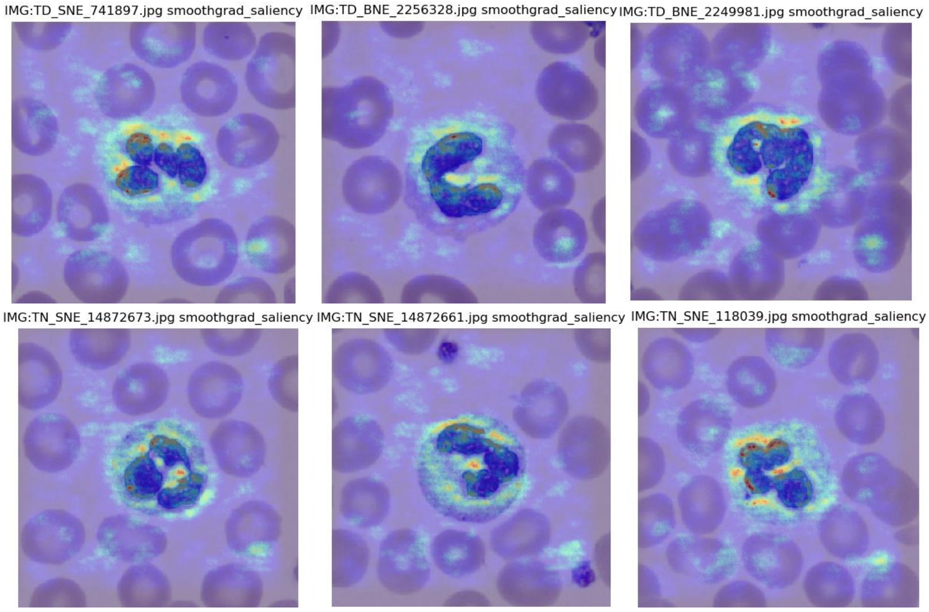

IMG:TD_SNE_741897.jpg smoothgrad_saliency
IMG:TD_BNE_2256328.jpg smoothgrad_saliency
IMG:TD_BNE_2249981.jpg smoothgrad_saliency
IMG:TN_SNE_14872673.jpg smoothgrad_saliency
IMG:TN_SNE_14872661.jpg smoothgrad_saliency
IMG:TN_SNE_118039.jpg smoothgrad_saliency

Figure 27. SmoothGrad Saliency maps for Dysplastic Neutrophils (top) and Normal Neutrophils(Bottom)

Regarding the Nucleus, the areas that present the highest saliency for normal cells are those that contain a high density deeply stained region. As for dysplastic cells, there is no clear pattern of highlighting the nucleus. Nonetheless, eight out of the ten dysplastic cells that present nuclear appendixes Figure 28. Are highlighted. This pattern is not observed on normal cells where only one of the nine cells that present this type of nuclear appendix is highlighted.

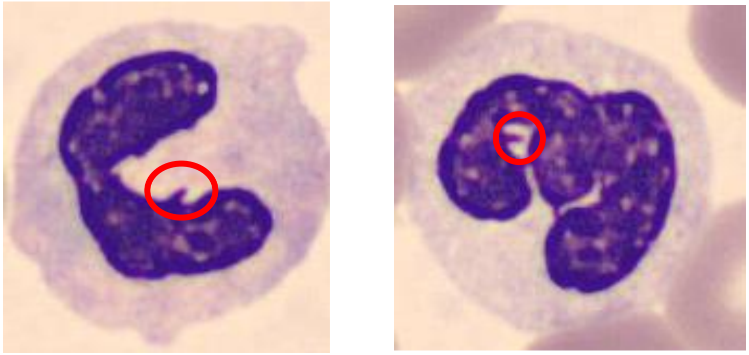


Figure 28. Dysplastic Nuclear Aberrations

Unfortunately, this visualization does not provide information regarding the number of lobes or the shape of the nucleus. Additionally, the scoring system used by the annotation application to determine the grade of granularity/agranularity of the cytoplasm as well as the heterogeneity/homogeneity of the nucleus cannot be directly correlated to the saliency output as saliency maps highlight localized regions on each of the nuclear structures and the scoring system used for the annotations evaluates the structures as a whole.

5.6. Occlusion Sensitivity Techniques

5.6.1. OSM: Occlusion Sensitivity Map

The algorithm applied to the interpretability dataset has been modified from the original paper to occlude the regions with the average color of the patched region instead of with a grey patch. As can be observed, replacing the patch color with the average value of the occluded region has a significant improvement in the output. The score for the background is kept constant and equal to the benchmarked score of the image. The four analysis have been performed on each image varying the patch size: 5x5px, 10x10px,20x20px,40x40px.

These output maps are interpreted the following way: The background value of the heat map matches the benchmarked score obtained for all the cells. Although the color from the background changes from one patch size to the other, for the same image, the score value is maintained constant. The heatness scale for the representation has been adjusted to the maximum and minimum scores obtained for each patch analysis. Regions marked with a higher heat, red, contribute in a positive matter to the predictions of the score. Instead lower heat or coldness, blue, contribute negatively to the prediction.

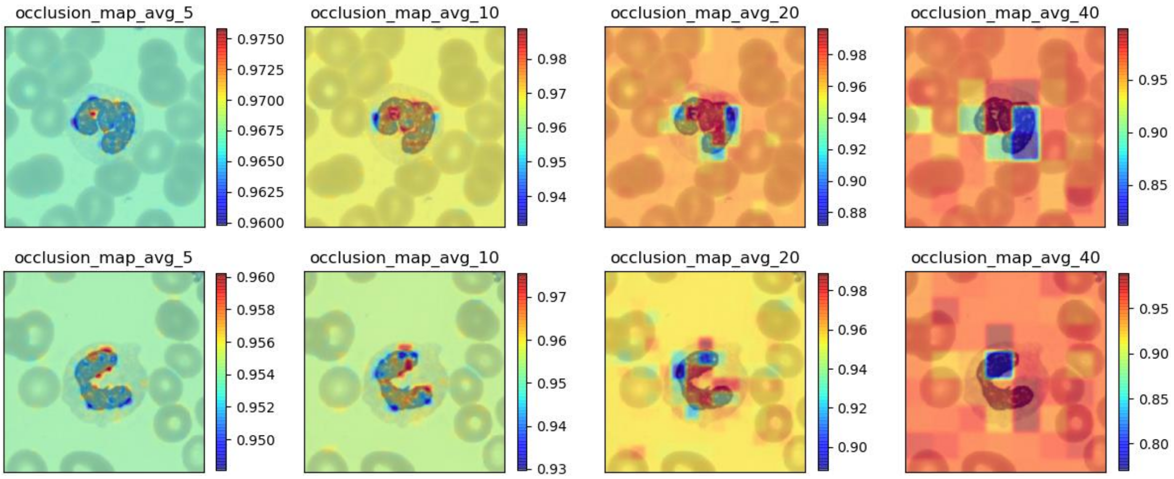


Figure 29. OSM average patch 5px, 10px, 20px, 40px [TD_BNE_2249981.jpg (Top) TD_BNE_2256328.jpg (Bottom)]

For dysplastic cells, the border areas of the nucleus are significant areas to determine the classification as there is a decrease in the score when occluded. Interestingly enough, when the occlusion is located anywhere inside the nucleus, there is an increase in the scoring output. This phenomenon is partly given, since the NN is lighter color region homogenous regions in the nucleus, indicating heterogenous tinction of the chromatin. This is described in section 1.4.4.

Morphology of a Dysplastic Neutrophils where a characteristic of dysplastic neutrophils is the presence of clumpy heterogeneous chromatinic regions in the nucleus.

Another feature worth noticing in the detection of dysplastic cells is the fact that there is no score change when the occluded region is located on the cytoplasm. This phenomenon is attributed to the lack of granules in the cytoplasm; therefore, when the average is calculated visually, the patch shows no difference to the region it is occluding. This argument is reinforced by the fact that the score slightly improves when any small remaining granule is occluded.

On the other hand, OSM does not report a significant score impact for normal cells (Figure 30). The patches hardly affect the output score, as can be observed in Table 4. On average, the 40px patches, being the most aggressive occlusion, only affect the score in a ±0.5%. However, the regions where the occlusions have the largest effect are the ones covering cytoplasm granules. Meaning the presence of granules is one of the most important features for classifiying neutrophils as normal.

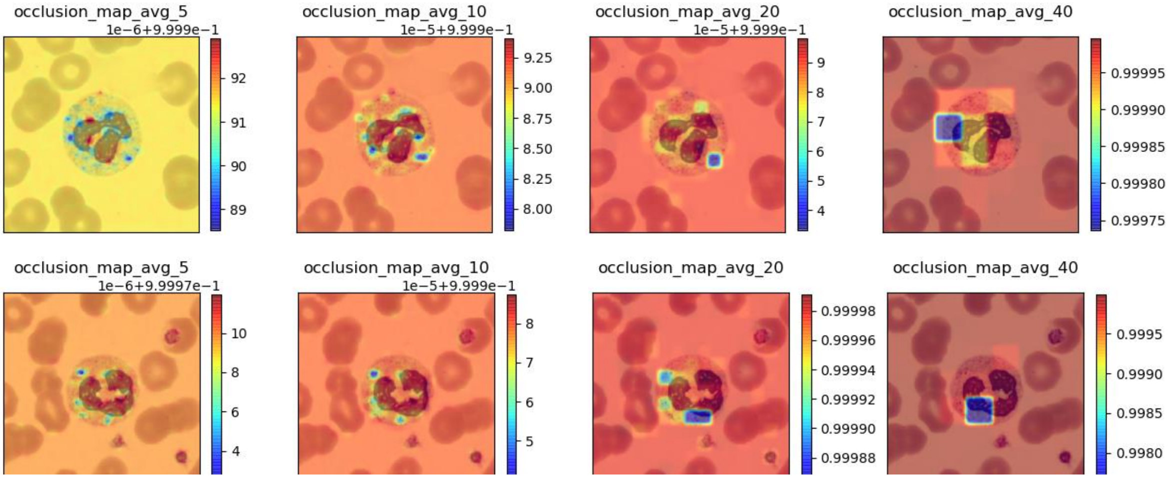


Figure 30. OSM average patch 5px, 10px, 20px, 40px [ZN_SNE_4198926.jpg (Top) ZD_BNE_5456886.jpg (Bottom)]

Table 4. contains the summary values for the OSM employing the average patch. This table has been acquired by analyzing dysplastic and normal neutrophils separately. For each image, the maximum and minimum patch scores are recorded into an array. The average score reduction is the average of the minimum patch score array. The average score increase is the average of the maximum patch score array. The maximum score reduction is the maximum score increase of all patches. The minimum score reduction is the maximum score reduction of all patches.

Patch size (px2)	Clas s	Avg. score reduction	Avg. score increase	Max. score reduction	Max. score increase
5	0	-0.3550%	0.3594%	-2.2708%	1.6703%
10	0	-1.5539%	1.5168%	-10.7401%	10.0465%
20	0	-3.9493%	2.0858%	-23.3115%	11.9172%
40	0	-6.6133%	3.0333%	-39.2397%	22.1266%
5	1	-0.0077%	0.0029%	-0.1633%	0.0612%
10	1	-0.0492%	0.0144%	-1.0655%	0.3202%
20	1	-0.1137%	0.0240%	-2.3905%	0.5240%
40	1	-0.5556%	0.0312%	-7.9618%	0.6764%

Table 4. OSM summary table

As reinforced by the previous visual analysis, the Normal-type Neutrophil prediction is not affected by the occlusion of small features. This means that the model relies upon a combination of geometries and features to diagnose a normal cell. Instead, Dysplastic type neutrophil detection is heavily influenced, firstly, by the lack of cytoplasmic granules and, secondly, by the presence of clumped chromatinic filaments in the nucleus.

5.6.2. LIME: Local interpretable model-agnostic explanations

The basic principle of this method is the same as for OSM, regions from the input image are occluded and the output score is evaluated independently for each patched region.

These patches have been generated with the quick shift algorithm. The following parameters have been adjusted to: *Ratio*= 0.5, this parameter [0,1] balances the color-space proximity image-space proximity, higher values give more weight to the color-space proximity. *Kernel_size*=2 width of the gaussian kernel used in smoothing the sample density. Higher values mean fewer clusters. *Max_dist*=100 cut-off point for data distances. Higher values mean fewer clusters. The average number of superpixels created per image is 227.Each image is iterated 700 times with different patched occluded randomly on each iteration.

The top 10 superpixels that contribute the most to the class classification are plotted for each image and masked over the original input. The result can be observed with Figure 31. The top row contains dysplastic neutrophils and the bottom row normal neutrophils. The yellow color highlights regions that contribute positively to the class prediction. On the other hand, the purple color indicates that the patch contributes negatively towards the class score.

As can be observed, most of the superpixels that have a more significant effect on the prediction are located on the nucleus for both classes. This general trend is also observed in the rest of the dataset for dysplastic neutrophils (Annex C5).

When analyzing the Dysplastic images, most of the superpixels that contribute positively towards the classification are yellow and are located on the nucleus. In contrast, most of the superpixels located on the cytoplasm contribute negatively to the class prediction. For the most part, this is due to the fact that those specific occluded regions contain a slight cytoplasmic granularity.

As for the Normal cell images, the opposite trend is observed. The superpixels that contribute positively to the classification are located in the cytoplasm regions where the granularity is the most present. The very few superpixels that contribute negatively to the class score are located on the nucleus. Averaging the value of the cytoplasm eliminates the granules, and the lack of granules is a symptom of dysplastic neutrophils.

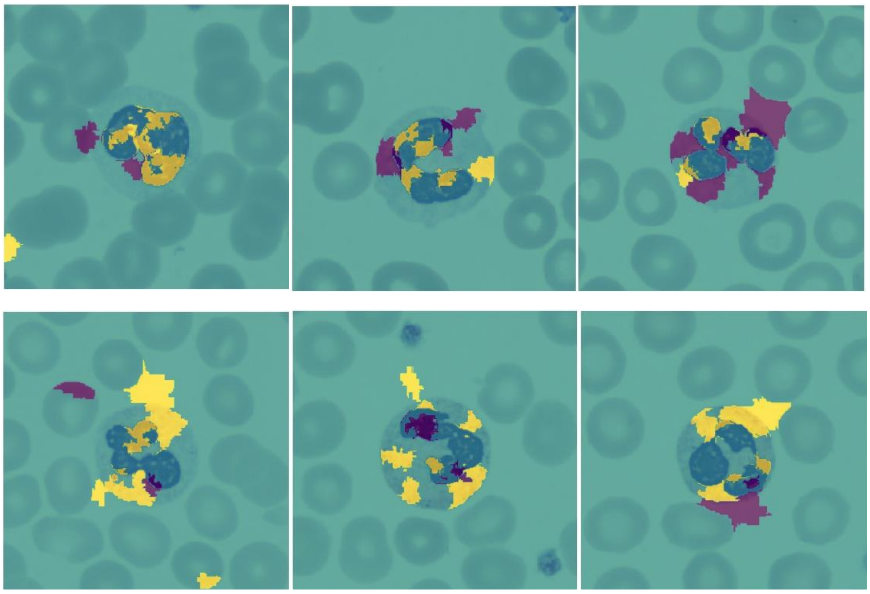


Figure 31. Lime Results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg)

Nonetheless, a consideration must be taken into account when analyzing these images. Occasionally regions located nowhere close to the cell are marked as positively or negatively contributing to the class prediction. These pixels might not have any significance towards the classification and have been highlighted due to the iterative process the method is based upon. Several pixels can be active on each iteration, and each pixel is activated numerous times through the analysis. There is a chance that superpixels that do not contribute to the prediction are activated numerous times with pixels that do. As the overall score of the iteration is evaluated, a non-realistic score might be attributed to those non-significant superpixels.

This analysis has shed light on an interesting situation. Given the iteration where most of the Neutrophil is occluded, DisplaciNet is unable to locate any relevant feature and attributes a perfect score of 1, Normal cell, regardless of the class of the cell.

5.7. Class Activation Map (CAM) Techniques

5.7.1. Grad-CAM

The results obtained from the application of GradCAM to the interpretability dataset can be observed in Figure 32. For normal cells, the method has highlighted the cytoplasm as the most relevant region for 23 out of the 23 images. For dysplastic cells, 19 out of the 23 images, the nucleus has been highlighted as the most relevant region of the image. For the remaining 4 images, the method has highlighted the cytoplasm. When this information is contrasted with the pathologist annotation, these four neutrophils present mild granularity. However, this method also attributes relatively high importance to all of the background red cells.

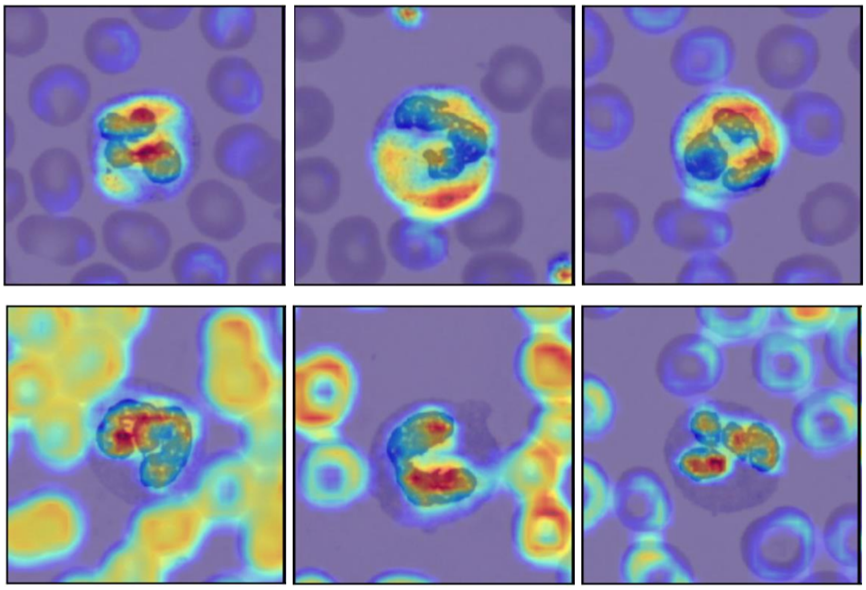


Figure 32. Grad-CAM results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg)

The results obtained from the application of GradCAM++ to the interpretability dataset can be observed in Figure 33. The results obtained are not satisfactory, as these should not vary too much from the simpler predecessor method. As can be observed, the highlighted regions lack sense if the output maps of the various images is observed as a whole. For some of the images, the image's background has been highlighted as the most relevant region. In other images, the most critical highlighted regions are the background red cells. A general trend is not present when comparing normal cells to dysplastic cells.

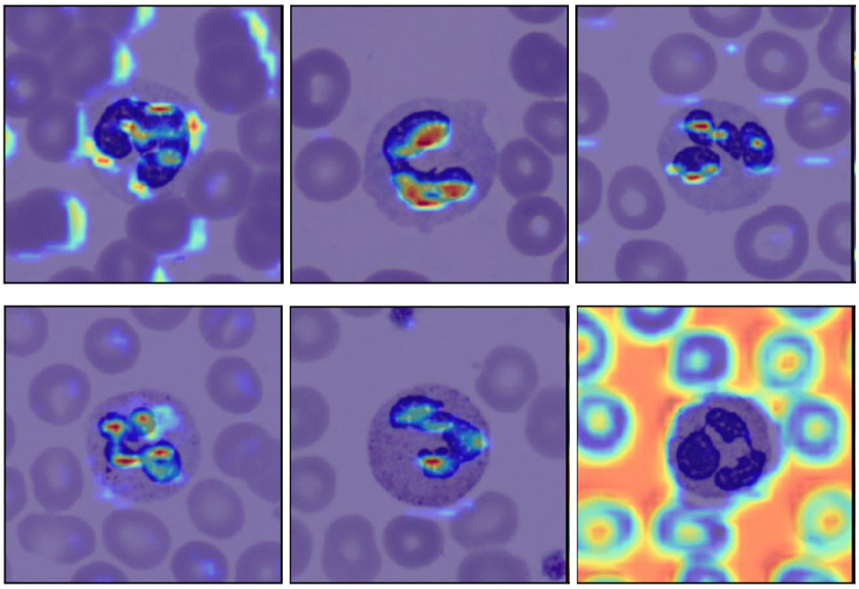


Figure 33. Grad-CAM++ results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg)

As a final remark, gradient-based methods have several issues, explained in previous methods such as saliency maps. These tend to add noise to the final result and pose a saturation issue when the model has been trained either with Sigmoid or ReLU activation functions. Another issue with this method is the False Confidence given by the linear combination of the different feature maps during the last stage [36]. Wang *et al.* demonstrated that occasionally activation maps with higher weight presented a lower contribution to the network's output than a zero baseline. This phenomenon was attributed to the global pool operation in combination with the gradient vanishing issue.

5.7.2. Score-CAM

The application of ScoreCAM to the interpretability dataset did not return satisfactory results. Out of the 56 maps obtained, 19 do not contain any information; the score map obtained is equal to zero. Coincidentally the 19 images that did not obtain any result are dysplastic cells. The remaining 37 images do not present any difference between dysplastic cells and normal cells, as shown in Figure 34, where the top row corresponds to dysplastic cells and the bottom row to normal cells. The result for all of these images is a highly localized region on the different lobes of the nucleus.

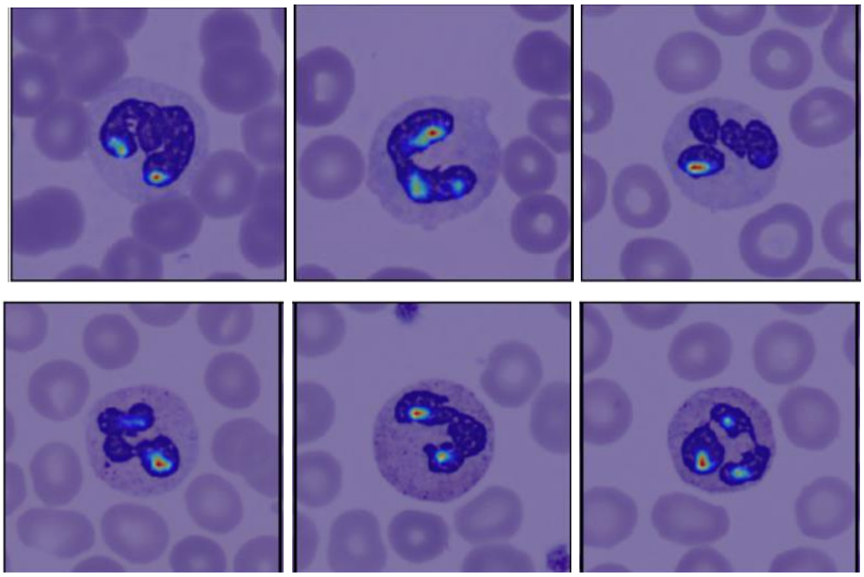


Figure 34. Score-CAM results (Top: TD_BNE_2249981.jpg, TD_BNE_2256328.jpg, TD_SNE_741897.jpg,; Bottom: TN_SNE_118039.jpg, TN_SNE_14872661.jpg, TN_SNE_14872673.jpg)

This method is partially based on an occlusion method where the superpixels of the different occlusion are obtained via the feature maps. The regions which are not highlighted in the feature map are entirely blacked out. This gives a partial explanation of why this method does not return the expected results, and as an occlusion method, it has been discussed in section 5.6. The substitution of an area by a black or gray patch returns inconsistent results. In addition, when the net is unable to locate a neutrophil in the image, the output score of the model is 1. For dysplastic features, the score is calculated as $Score_{image} = 1 - Prediction_{image}$ as the net has not been able to locate the neutrophil, the output score is 0. The ReLU activation function truncates any negative contributing features, explaining why many of the dysplastic features do not receive any score.

5.8. Background Generalisation Test

The final test performed on DisplaciaNet is a background independence test. With this test, we aim to prove that the background does not influence the scoring prediction of the model. The scores for the 56 original images and the 56 segmented images can be visualized in the Annex D. Results of the Background Generalization Test. Table 5 contains the summary of scores for both sets of images divided by class and the summary of the differences between scores.

	Original Image		Masked Image		Difference			
	Avg. Score	S. D	Avg. Score	S.D.	Avg. Difference	S.D	max	min
Dysplastic	0.0422	0.0920	0.0606	0.1337	-0.0184	0.1053	0.1986	-0.4485
Normal	0.9996	0.0017	0.9995	0.0013	0.0001	0.0022	0.0057	-0.0082
Wrong dysplastic	0.9977	0.0044	0.9224	0.1034	0.0753	0.0994	0.2511	0.0000
Wrong normal	0.2696	0.2139	0.4874	0.1850	-0.3638	0.0880	-0.2651	-0.4891

Table 5. Summary of the background generalization test

As can be appreciated by the average difference column, the background artifacts have minimal effect on the correctly classified images. For the Normal Neutrophils, the score difference is neglectable. As for Dysplastic Neutrophils, the average score change is 0.1%, with the exception of image ZD_SNE_2759718.jpg(Figure 35), whose score changed from 0.049 to 0.497. As can be appreciated in the image, this cell presents hypogranularity and hypolobulation, typical features of a dysplastic cell, but also an extremely Homogeneous nucleic chromatin, typical features of a normal cell.

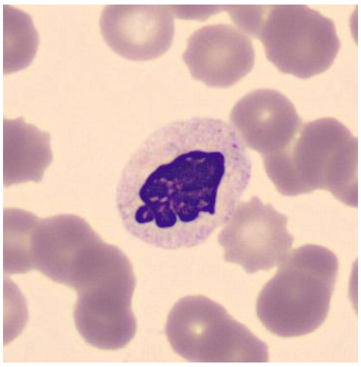


Figure 35. Dysplastic Neutrophil (ZD_SNE_2759718.jpg) source: [6]

As for the wrongly classified images, there is a tremendous change in the ground truth Normal Neutrophil score. The average change is 36% towards the correct class, even classifying one of the cell images correctly.

6. Discussion

This chapter focuses on comparing the different results obtained for the various Deep Learning Interpretability techniques. Homogeneity is expected in order to deem the study successful.

The convolutional blocks of Dysplacianet, are the backbone to the classification. These blocks filter out any unnecessary information to perform the classification from the input image. As such, understanding what features are being maintained endows us with a slight overview of the decision criteria of the CNN. By employing the algorithm t-SNE, we can replicate the feature distribution of the high dimensional convolutional block output onto a 2-dimensional space. From the plot acquired, at first glance we can clearly identify one big cluster, this cluster represents the Dysplastic Neutrophils. A second less prominent cluster, is noticeable and groups the Normal Neutrophils. T-SNE performs a non-lineal dimensionality reduction, meaning that the distances between ploted points, represent semantic relations between images in real life.

From this low dimensional feature representation, we can extract that the convolutional blocks are able to extract relevant features from the input images. This statement is backed up by the analysis of the different feature maps of the convolutional blocks.

By analyzing the different feature maps, we can slightly grasp what features the CNN is highlighting for each image. This analysis is extremely tedious and seldom reports quantitative data. Firstly, the different feature maps are analyzed independently, we are able grasp a slight knowledge of what some of the different filters highlight. For the last convolutional layer, we can clearly see feature maps that highlight the whole cytoplasm, and others selective areas of the cytoplasm were a pinkish granules are present. Other feature maps highlight the cytoplasm or regions where there are high density chromatin contents. By comparing the Dysplastic image feature maps against the Normal image feature maps. It is evidenced that certain feature maps have a higher activation for one class or the other class. This indicated that the network is able to identify key features of both classes

Two different Methods of Saliency Maps were applied, Vanilla Saliency an Smoothgrad Saliency. Vanilla saliency did not report any valuable information. Instead, the results for the SmoothGrad saliency maps returned information that is in accordance with the pathologist workflow. For both classes the region with the highest interest for the classification is the cytoplasm. For the Pathologists the main indicator of Dysplasia in cells is the lack of granules in the cytoplasm.

Nonetheless, saliency maps do not offer quantitative information. The information extracted is only qualitative.

Another interesting feature highlighted by the saliency maps for dysplastic cells, are some nuclear appendixes. After contrasting this information with the expert pathologist, it was understood that this feature is not representative of the Dysplasia disease. These appendixes are known as the Barr bodies. The chromatinic agglomeration are caused by the sexual chromosome, chromosome 23, and can only be developed in the female biologic gender. It is characterized by the inactivity of one of the two X chromosomes and is usually represented in the last lobe of the nucleus. The theory behind the net detecting this particular feature is that this chromatinic agglomeration is present in most neutrophils of a single patient. Because the net has been trained with complete blood smears, a high number of smears that contained this appendix have been labeled as dysplastic in the dataset, thus biasing the net to learn how to identify this feature.

Occlusion sensitivity maps, Gradcam and Lime returned coherent results. For the three techniques the cytoplasm was labeled as the most important region for the Dysplastic cells. On the other hand, the most important region for Normal cells is the nucleus. This is evidenced by the lack of granules in the cytoplasm, and therefore the CNN must turn its focus toward the nucleus.

A very interesting consequence of applying small patches (5x5px and 10x10px) to OSM, is the ability to alter so slightly features on the different parts of the cells. When punctual granules of the cytoplasm are eliminated, the output score decreases for Normal Neutrophils, this event is in agreement with the pathologist reasoning. On the other hand, when the patches are located aver the nucleus for Dysplastic cells, it performs a smoothing effect on the chromatinic content. These modifications are interpreted by the net as a more homogeneous nucleus, characteristic of normal cells, and therefore the output score also decreases.

The capabilities of generalizing the background, are backed up by the background generalization test and the OSM. For the generalization test, removing the background in correctly classified images had a negligible effect on the output score. This generalization is also supported by the OSM analysis. The occlusions cover the background artifacts had little effect on the output score.

Summing up, DisplasiaNet mainly focuses on the granularity of the cytoplasm to classify Normal cells. The chromatinic density is evaluated by DisplasiaNet regarding if it presents a homogeneous density or a heterogenous density. Unfortunately, no algorithm applied could provide any information regarding the lobular segmentation.

7. Conclusion and Future work

The aim of this project was to perform a Deep Learning Interpretability study on the Neural Network DisplasiaNet. This particular CNN is specialized in the classification of Dysplastic and Normal Neutrophil images obtained from peripheral blood smears. By applying interpretability techniques, insights into the inner works of the CNN have been revealed. This information has allowed us to answer the questions raised in the introductory chapter.

- ***Does the NN focus on the same features as the pathologist***?
  Yes. DisplaciaNet's first focus is to analyze the granularity of the cytoplasm. If the cytoplasm presents high granularity the image is labeled as Normal. The second priority of the model is to analyze the nucleus, when the image presents mild granularity. The chromatinic density as the nucleus segmentation decides whether the cell is Dysplastic or not. If the Cell image does not present granularity, the Neutrophil is classified as dysplastic.
- ***Does the NN evaluate the features in the same matter?***
  Yes. The criterion for evaluation is the same as the expert pathologists use. Nonetheless this specific study has not been able to find a relation between the proposed scoring system presented in the Cell Annotatio APP and the scoring systems of DisplasiaNet
- ***Is the NN able to generalize?***
  Yes. The different methods applied show that for correctly classified images the background of the cell has a neglectable influence on the output score.
- ***Moreover, can the wrongly classified images provide information on how to improve the NN?***
  The ultimate reason for why the CNN misclassifies certain images is not clear. However, for misclassified Normal Neutrophil images the background artifacts have a strong influence on the classification Output. Also mislabeling in the training dataset has been revealed.

The answers to these questions contribute to deem the NN with Transparency, Explainability, and Reliability as the ultimate goal is to build enough trust in order to be able to integrate such a tool into the medical diagnosis workflow.

Nonetheless, interpretability of deep learning models is a complicated task. Multiple parallel analysis must be performed in order to make a strong case to defend/understand the decision criteria. The “no free lunch” theorem applies to DL interpretability meaning that one technique is not enough to shed light on all the decision criterion of the model. Instead, the overall picture is painted by comparing and contrasting the different obtained results.

As future work we suggest the creation of a whiter-box model to be executed alongside DysplaciaNet. This whiter-box model, would mimic the approach the expert pathologist follows to diagnose Dysplasia in Neutrophils. Firstly, a CNN should be trained to segment the cytoplasm and the nucleus from the image. Then a trained regression model would evaluate the granularity for the cytoplasm and the chromatinic density of the nucleus. These regression outputs would be treated as a score. The addition of both scores in combination with a binary threshold would be able to diagnose the Neutrophils. The final cell diagnosis would be achieved by comparing the final classification of DisplasiaNet with the one from the Whiter-Box model. This methodology would allow to identify mismatched diagnosis as well as endowing the pathologist with contestability.

The tool to create the dataset for this approach, Cell Annotation APP, is one of the contributions of this project. It allows for the creation of segmentation masks for the nucleus and the cytoplasm as well as it offers a scoring system to train the regressions.

8. References

[1] K. Hornik, "Approximation capabilities of multilayer feedforward networks.," *Neural Networks,* vol. 4, no. 2, pp. 251-257, 1991.

[2] Zhang Z, Beck MW, Winkler DA, Huang B, Sibanda W, Goyal H, "Opening the black box of neural networks: methods for interpreting neural network models in clinical applications.," *Ann Transl Med.,* vol. 6, no. 11, p. 216, 2018.

[3] L.Malcovati and M.Cazzola , "Myelodysplastic syndromes - Coping with ineffective hematopoiesis.," *New England Journal of Medicine,* vol. 352, no. 6, pp. 536-538, 2005.

[4] R. Shouval, J. A. Fein, B. Savani, M. Mohty, and A. Nagler, "Machine learning and artificial intelligence in haematology," *British Journal of Haematology,* 2020.

[5] A. A. Lipes, "Deep Learning System for the Automatic Classification of Normal and Dysplastic Peripheral Blood Cells as a Support Tool for the Diagnosis (Doctoral dissertation)," Universitat de Barcelona, Barcelona, 2021.

[6] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a "right to explanation"," *AI Magazine,* vol. 38, 2017.

[7] U. K. G. H. R. &. G. N. Germing, " Myelodysplastic syndromes: diagnosis, prognosis, and treatment.," *Deutsches Arzteblatt international,* vol. 110, no. 46, p. 783–790, 2013.

[8] M. D. C. F. Ramos Ortega, "SÍNDROMES MIELODISPLÁSICOS.," in *Pregrado de Hematologia, 4º edicion,* Madrid, Luzán 5, 2017, pp. 318-320.

[9] N. C. I. (US), "Chronic Lymphocytic Leukemia Treatment(PDQ)," in *PDQ Cancer Information Summaries*, Bethesda, National Institutes of Health, 2002.

[10] D. A. Arber, A. Orazi, R. Hasserjian, et al., "The 2016 revision to the World Health Organization classification of myeloid neoplasms and acute leukemia," *Blood,* vol. 127, no. 20, p. 2391–2405, 2016.

[11] J. K. P. Actor, "2 - Cells and Organs of the Immune System," in *Elsevier's Integrated Review Immunology and Microbiology (Second Edition),* W.B. Saunders, 2012, pp. 7-16.

[12] P. Veda, "Why are neutrophils polymorphonuclear?," vol. 9, no. 2, 2011.

[13] J. B. C. Niels Borregard, "Granules of the Human Neutrophilic Polymorphonuclear Leukocyte," *Blood,* vol. 89, no. 10, pp. 3503-3521, 1997.

[14] A. Acevedo, A. Merino, S. Alferez A. Molina, L. Boldú J. Rodellar, "A dataset for microscopic peripheral blood cell images for development of automatic recognition systems," *Mendeley Data,* p. V1, 2020.

[15] A. Maerino, "Cap. 6 Síndromes Mielodisplásicos y Neoplasias Mielodisplásicas/Mieloproliferativas," in *Manual de Citología de sangre periférica y líquidos biológicos*, Madrid, Panamericana, 2019, pp. 8,9.

[16] Acevedo, A, Merino, A, Boldú, L, Molina, A, Alférez, S, and Rodellar, J., "A new convolutional neural network predictive model for the automatic recognition of hypogranulated neutrophilsin myelodysplastic syndromes.," *Computers in Biology and Medicine,* vol. 134, no. 104479, 2021.

[17] T. Sejnowski, "The unreasonable effectiveness of deep learning in artificial intelligence.," *Proceedings of the National Academy of Sciences,* vol. 117, no. 48, pp. 30033-30038, 2020.

[18] Zachary C. Lipton, "The Mythos of Model Interpretability," *Communications of the ACM,* vol. 61, no. 10, pp. 36-43, October 2018.

[19] P. T. A. L. a. K. T. Yu Zhang, "A Survey on Neural Network Interpretability," *arXiv,* no. 2012.14261v3, 2021.

[20] R. Roscher, B. Bohn, M. F. Duarte and Jochen Garcke, "Explainable Machine Learning for Scientific Insights and Discoveries," *IEEE Access,* vol. 1, no. 1, 2020.

[21] A. B. Arrieta, N Díaz-Rodríguez, J. Del Ser, A. Benneot, S.tabik, A. Barbado, S.Garciía, S. Gil-Lopez, D. Molina, R. Benjamins et al., "Explainable Artificial intelligence(xai): Concepts, Taxonomies, opportunities and challenges towards responsible AI.," *Information Fusion,* vol. 58, 2020.

[22] F. -L. Fan, J. Xiong, M. Li and G. Wang, "On Interpretability of Artificial Neural Networks: A Survey.," *IEEE Transactions on Radiation and Plasma Medical Sciences,* 2021.

[23] F. Doshi-Velez and B. Kim, "Towards a Rigorous Science of Interpretable Machine Learning," *arXiv,* no. 1702.08608v2, 2017.

[24] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," *Proceedings of the IEEE Conference on Computer Vision and Patters Recognition,* pp. 427-436, 2015.

[25] Center for Devices and Radiological Health, Center for Biologics Evaluation and Research, Center for Drug Evaluation and Research, "Clinical Decision Support Software," Food and Drug Administration, 2019.

[26] European Parliament, Council of the European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC," Official Journal of the European Union, 2016.

[27] L. Breiman, J.hH. Friedman, R.A. Olshen, C.. Stone, Classification And Regression Trees, Boca Raton: Routledge, 1984.

[28] N. Altman, "An Introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician,* vol. 46, no. 3, pp. 175-185, 1992.

[29] C. Sammut, G.I. Webb, Encyclopedia of Machine Learning., Boston: Springer, 2011.

[30] F. Chollet, "Visualizing what covnets learn," in *Deep Learning with Python*, Shelter Island, Manning, 2018, pp. 160-177.

[31] D. Bau, B. Zhou, A. Khosla, A. Oliva and A. Torralba, "Network Dissection: Quantifiying interpretability of deep visual representations," *CVPR,* 2017.

[32] Y. Li, j. Yosinski, J.clune, H,Lipson and J.E. Hopcroft, "Convergent Learning: Do different Neural Networks learn the same representations?," *ICLR,* 2016.

[33] B. Zhou, A. Khosla, A.Lapedriza, A. Oliva, Aude and A. Torralba, "Object detectors emerge in Deep Scene CNNs," *ArXiv: Computer Vision and Pattern Recognition,* 2014.

[34] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision-ECCV 2014*, Springer, 2014, p. 818–833.

[35] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization.," *Proceedings of the IEEE conference on computer vision and pattern recognition,* pp. 2921-2929, 2016.

[36] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations rom deep networks via gradient-based localization.," *Proceedings of the IEEE International Conference on Computer Vision,* pp. 618-626, 2017.

[37] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," *Procedings of the IEEE Winter Conference on Applications of Computer Vision,* pp. 839-847, 2018.

[38] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, X. Hu, "Score-CAM: Improved Visual Explanations Via Score-Weighted Class Activation Mapping," *arXiv: Computer Vision; Machine Learning,* no. arXiv:1910.01279v1, 2019.

[39] A. Bansal, A. Farhadi and D. Parikh, "Towards transparent systems: Semantic characterization of failure models.," *ECCV,* 2014.

[40] Y. Wang, H. Su, b. Zhang and X. hu, "Interpret Neural Networks by identifiying critical data routing paths," *CVPR,* 2018.

[41] K. Simonyan, A. Vedaldi and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," *arXiv : Computer Vision and Pattern Recognition,* no. arXiv:1312.6034v2, 2014.

[42] D. Smilkov, N. Thorat, B. Kim, F. Viégas and M. Wattenberg, "SmoothGrad: removing noise by adding noise," *arXiv: Machine Learning; Computer Vision and Pattern Recognition,* no. arXiv:1706.03825, 2017.

[43] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin, "Why should I trust you?: Explaining the predictions of any classifier.," *In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining,* p. 11351144, 2016.

[44] Scott M Lundberg and Su-In Lee, "A Unified Approach to Interpreting Model Predictions," in *31st Conference on Neural Information Processing Systems (NIPS 2017*, Long Beach, CA, USA, 2017.

[45] l. Shapley, "A value for n-person games," *Contributions to the theory of games. Princeton University Press,* pp. 307-317, 1953.

[46] P. Van Molle, M de Strooper, T. Verbelen, B. Vankeirsblick, P. Simoens and B. Dhoedt, "Visualizing convolutional neural networks to improve desicion support for skin lession classification," *Understanding and Interpreting Machine Learning in Medical Image computing Applications,* pp. 115-123, 2018.

[47] A. R. J. E. S. W. A. N. J. C. U. E. B. L. C. a. G. D. A. Zaritsky, "Interpretable deep learning uncovers cellular properties in label-free live cell images that are predictive of highly metastatic melanoma,," *Cell Systems,* vol. 12, no. 7, pp. 733-747. e6, 2021.

[48] J.R. Zech, M.A. Bagdeley, M. Liu, A.B. CostaJ.J. Titanoand E.K. Oerman, "Variable generalisation performaceof a deep learning model to detect pneumonia in chest radiographs: A cross sectional study," *PLooS Medicine,* vol. 15, no. 11, 2018.

[49] A. A. Cruz-Roa, J. E. Arevalo Ovalle, A. Madabhushi, F. A. González Osorio, "A Deep Learning Architecture for Image Representation, Visual Interpretability and Automated Basal-Cell Carcinoma Cancer Detection," in *Medical Image Computing and Computer-Assisted Intervention -- MICCAI 2013,* Berlin, Springer Berlin Heidelberg, 2013, pp. 403-410.

[50] S. Pereira, R.Meier, R. McKinley, R. Wiest, V. Alves C.A. Silva and M.Reyes, "Enhancing intepretability of automatically extracted machine learning features: application to a RBM- Random forest systems on a brainlesion segmentation.," *Medical Image Analysis,* vol. 44, pp. 228-244, 2018.

[51] J. W. J. C. W. e. a. Diao, "Human-interpretable image features derived from densely mapped cancer pathology slides predict diverse molecular phenotypes," *Nature Communications,* vol. 12, no. 1613, 2021.

[52] Plotly, "Plotly Dash Documentation," [Online]. Available: https://dash.plotly.com/. [Accessed 20 04 2021].

[53] Docker, "Docker Documentation," [Online]. Available: https://docs.docker.com/. [Accessed 23 04 2021].

[54] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research,* vol. 9, pp. 2579-2605, 2008.

[55] L. v. d. Maaten, "Learning a Parametric Embedding by Preserving Local Structure," *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics,* vol. 5, pp. 384-391, 2009.

[56] A. Shrikumar, P Greenside, and A. Kundaje, "Learning important features through propagating activation differences.," *Proceedings of the 34th International Conference on Machine Learning,* vol. 70, 2017.

[57] Z. Zhong, L. Zheng, G. Kang, S. Li and Y. Yang , "andom Erasing Data Augmentation.," *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 34, no. 07, pp. 13001-13008, 2020.

[58] S. Yun, D. Han, S. Joon Oh, S. Chun, J. Choe, Y. Yoo, "CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features," *ArXiv: Computer Vision and Pattern Recognition,* 2019.

[59] A. Vedaldi,S. Soatto, "Quick shift and kernel methods for mode seeking," *European Conference on Computer Vision - ECCV 2008,* pp. 705-718, 2008.

[60] S. Hernandez, "Github Repository - Cell Annotation APP," [Online]. Available: https://github.com/Stevecreations/Cell_Annotation_TFM. [Accessed 2021 07 02].

[61] S. Hernandez, "Docker Repository - Cell Annotation APP," [Online]. Available: https://hub.docker.com/repository/docker/stevecreations/tfm_cell_annotation_app/. [Accessed 2021 07 02].

[62] S. Hernandez, "DysplaciaNet Github repository," [Online]. Available: https://github.com/Stevecreations/DysplaciaNet_Interpretability_Study.

[63] S. Hernandez, "Cell Annotation App," UPC, [Online]. Available: http://deepbox.eebe.upc.edu:8050/. [Accessed 2021 06 29].

[64] "Docker desktop suite," Docker INC., [Online]. Available: https://www.docker.com/products/docker-desktop. [Accessed 22 09 2021].

[65] werdlow SH, Campo E, Harris NL, Jaffe ES, Pileri SA, Stein H, Thiele J, WHO vClassification of Tumours of Haematopoietic and Lymphoid Tissues, 4th ed, International Agency for Research on Cancer, 2017.

[66] M.D. Zeiler and R. Fergus, "Visualizing and understanding convolutional Networks.," *ECCV,* pp. 818-833, 2014.

Annex A. Cell Annotation Application Instructions

The following Annex contains the detailed intructions of the Cell Annotation Application. These are visisble to the user when the About button, present in the header, is presed.

App description

Cell Dysplasia Annotation App enables the user to annotate microscopic images of blood cells. There are three main indicators pathologists use to detect whether a cell is dysplasic or not. These are:

a) Check for absence of granularity in the cell's cytoplasm. A healthy cell presents a certain granularity in the cytoplasm.

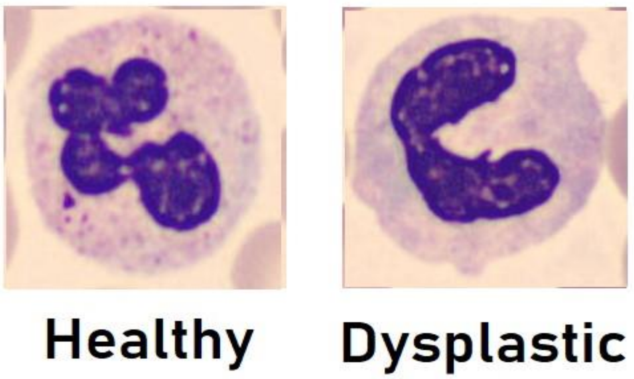
b) Check for Hyperchromasia. A dysplastic cell will display an increased chromatin content resulting in a deeply stained nucleus.

c) Number of lobes of cell nuclei. A healthy cell usually presents between 2 and 5 lobes.

Other factors to bear in mind are:

d) Ratio between the nuclei and the cytoplasm as Dysplastic cells present a larger nucleus than healthy ones

e) An increase of the Miotic figures.



A. 1 [healthy_vs_displastic_cell](/assets/images/healthy_vs_displastic.jpg)

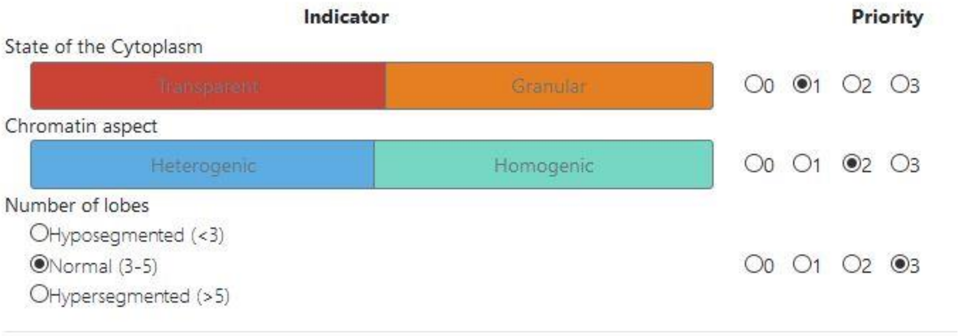
This app allows the user to annotate and classify areas in an image.

- Navigate the images in directory
- Freehand draw, using the mouse to highlight specific areas of the image.
- Select annotation type Cytoplasm(Granular / Transparent), Chromatin (Heterogenic / Homogenic), Nº of Lobes (Hyposegmented/ Normal / Hypersegmented)
- Select the priority of each characteristic when deciding if a cell is Healthy or presents Dysplacia.
- Export the annotations made as a txt file (json format) and as an image.

How to use this app

The first step is to update the path directoryehre the images are stores by clicking on `Update Image Path`.

To annotate the image, first select the annotation label you want to apply:



Indicator Priority

State of the Cytoplasm

Transparent Granular ○0 ●1 ○2 ○3

Chromatin aspect

Heterogenic Homogenic ○0 ○1 ●2 ○3

Number of lobes

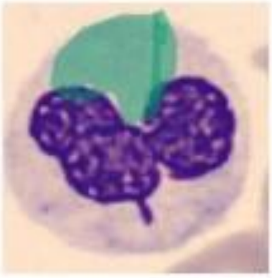
○Hyposegmented (<3)

●Normal (3-5) ○0 ○1 ○2 ●3

○Hypersegmented (>5)

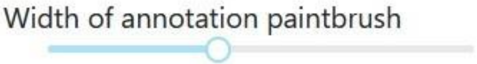
A. 2[Screenshot of label selector](/assets/images/select_label.jpg)

Then highlight the desired image area to annotate with the cursor:



A. 3![Screenshot of annotation](/assets/images/draw_annotation.jpg)

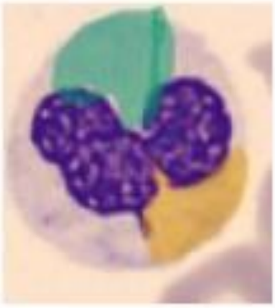
The width of the annotation brush can be changed with the slider bar `Width of annotation paintbrush` .



Width of annotation paintbrush

A. 4![Screenshot of width_paintbrush](/assets/images/width_paintbrush.jpg)

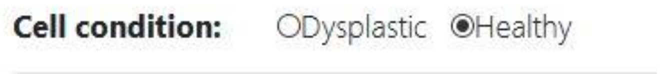
Additional annotations can be made by selecting a new label and highlighting the regions on the image.



A. 5![Screenshot of_additional_annotation](/assets/images/additional_annotation.jpg)

In order to classify the images correctly, it is important to select the decision criteria priority by clicking on the radio butons `priority - 0(null) / 1(min) / 2 / 3(max)`

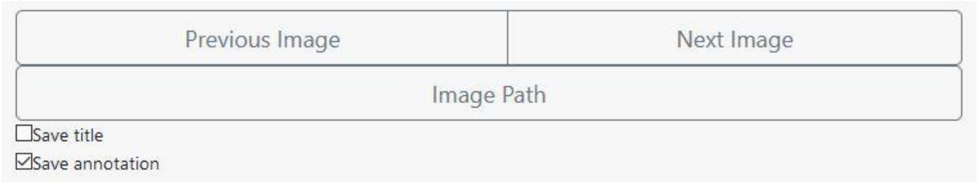
The user can also state if a cell is dysplactic or healthy by the redio button `Cell condition`



Cell condition: ○Dysplastic ●Healthy

A. 6 ![Screenshot of_Cell_condition](/assets/images/cell_condition.jpg)

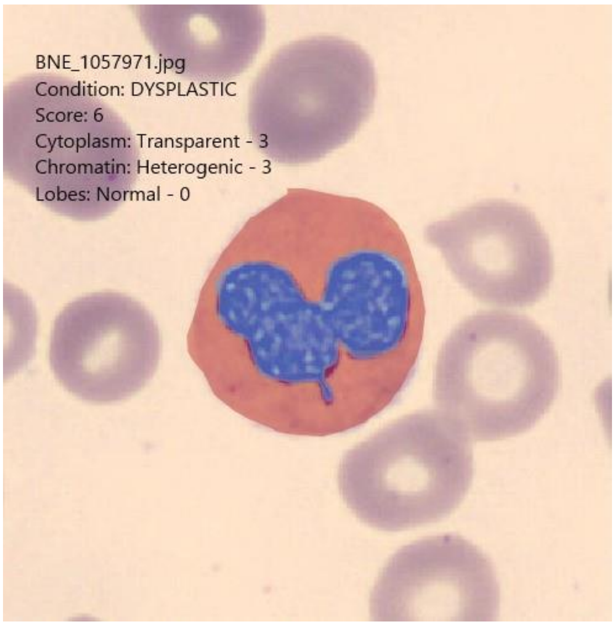
To select a different image the press on the `previous` and `next`. The current annotations made on the image will be stored in a txt file in the same directory as the image source. An image with the annotations will also be stored will aso be stored.



☐Save title
☑Save annotation

A. 7 ![Screenshot of_Navigation](/assets/images/navigation.jpg)

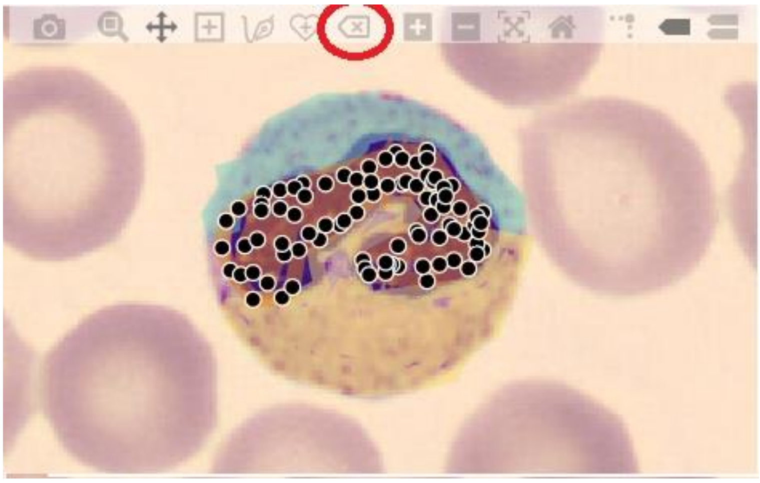
To download an image with the annotations and/or info entered press on the check boxes `Save annotation` `Save_title` respectively. The image below is an example of when both annotation and title are saved.



BNE_1057971.jpg
Condition: DYSPLASTIC
Score: 6
Cytoplasm: Transparent - 3
Chromatin: Heterogenic - 3
Lobes: Normal - 0

A. 8 ![Screenshot of_Save_title_Annotation](/assets/images/save_title_annotation.jpg)

To erase an already made annotation click of the desired annotation and then hit the errase button on the top of the image editor.



A. 9 ![Erase_annotation](/assets/images/errase_annotation.jpg)

Annotations can also be adjusted by draging the black circunferences of the border of the annotation to the desired location.

Credits

S. Hernandez - 2021 Universitat Politecnica de Catalunya in collaboration with Hospital Clinic de Barcelona

Annex B. Annotation Data + DisplasiaNet prediction

Image Name	Dataset Class	DisplasiaNet		Cytoplasm		Chromatin		Lobe		Total score	Pathologist Diagnosis
		Score	Prediction	State	Score	State	Score	State	Score		
TD_BNE_2249981	0	0.0334	0	Transparent	3	Hetero	3	Hyposegmented	1	7	DYSPLASTIC
TD_BNE_2256328	0	0.0463	0	Transparent	3	Hetero	2	Hyposegmented	1	6	DYSPLASTIC
TD_SNE_741897	0	0.0009	0	Transparent	3	Hetero	2	Normal	0	5	DYSPLASTIC
ZD_BNE_2433069	0	0.3919	0	Transparent	1	Hetero	2	Hyposegmented	1	4	DYSPLASTIC
ZD_BNE_2433092	0	0.0013	0	Transparent	3	Hetero	1	Hyposegmented	1	5	DYSPLASTIC
ZD_BNE_2433255	0	0.2043	0	Transparent	2	Hetero	3	Hyposegmented	1	6	DYSPLASTIC
ZD_BNE_2433280	0	0.0017	0	Granulated	-1	Hetero	3	Hyposegmented	1	3	DYSPLASTIC
ZD_BNE_2759813	0	0.0002	0	Transparent	3	Hetero	3	Normal	1	7	DYSPLASTIC
ZD_SNE_2433225	0	0.1755	0	Transparent	3	Hetero	-1	Hyposegmented	1	3	DYSPLASTIC
ZD_SNE_2543049	0	0.0619	0	Transparent	1	Hetero	1	Hyposegmented	1	3	DYSPLASTIC
ZD_SNE_2543060	0	0.0001	0	Transparent	3	Hetero	2	Normal	0	5	DYSPLASTIC
ZD_SNE_2543086	0	1.0E-07	0	Transparent	2	Hetero	2	Hyposegmented	1	5	DYSPLASTIC
ZD_SNE_2759718	0	0.0490	0	Transparent	2	Hetero	2	Hyposegmented	1	5	DYSPLASTIC
ZD_SNE_2759795	0	0.0037	0	Transparent	3	Hetero	2	Hyposegmented	1	6	DYSPLASTIC
ZD_SNE_6643852	0	4.6E-05	0	Transparent	3	Hetero	3	Normal	0	6	DYSPLASTIC
ZD_SNE_6643874	0	2.5E-06	0	Transparent	3	Hetero	2	Hyposegmented	1	6	DYSPLASTIC
ZD_SNE_6643949	0	0.0007	0	Transparent	3	Hetero	3	Hypersegmented	1	7	DYSPLASTIC
ZD_SNE_6700392	0	2.3E-06	0	Transparent	3	Hetero	3	Hypersegmented	1	7	DYSPLASTIC
ZD_SNE_6700394	0	2.2E-05	0	Transparent	3	Hetero	3	Normal	0	6	DYSPLASTIC
ZD_SNE_6700404	0	1.5E-06	0	Transparent	3	Hetero	3	Hypersegmented	1	7	DYSPLASTIC
ZD_SNE_6700428	0	4.3E-05	0	Transparent	3	Homo.	-1	Normal	0	2	DYSPLASTIC
ZD_SNE_6700434	0	0.0001	0	Transparent	3	Homo.	3	Normal	0	6	DYSPLASTIC
ZD_SNE_6700475	0	1.5E-05	0	Transparent	3	Homo.	3	Normal	0	6	DYSPLASTIC

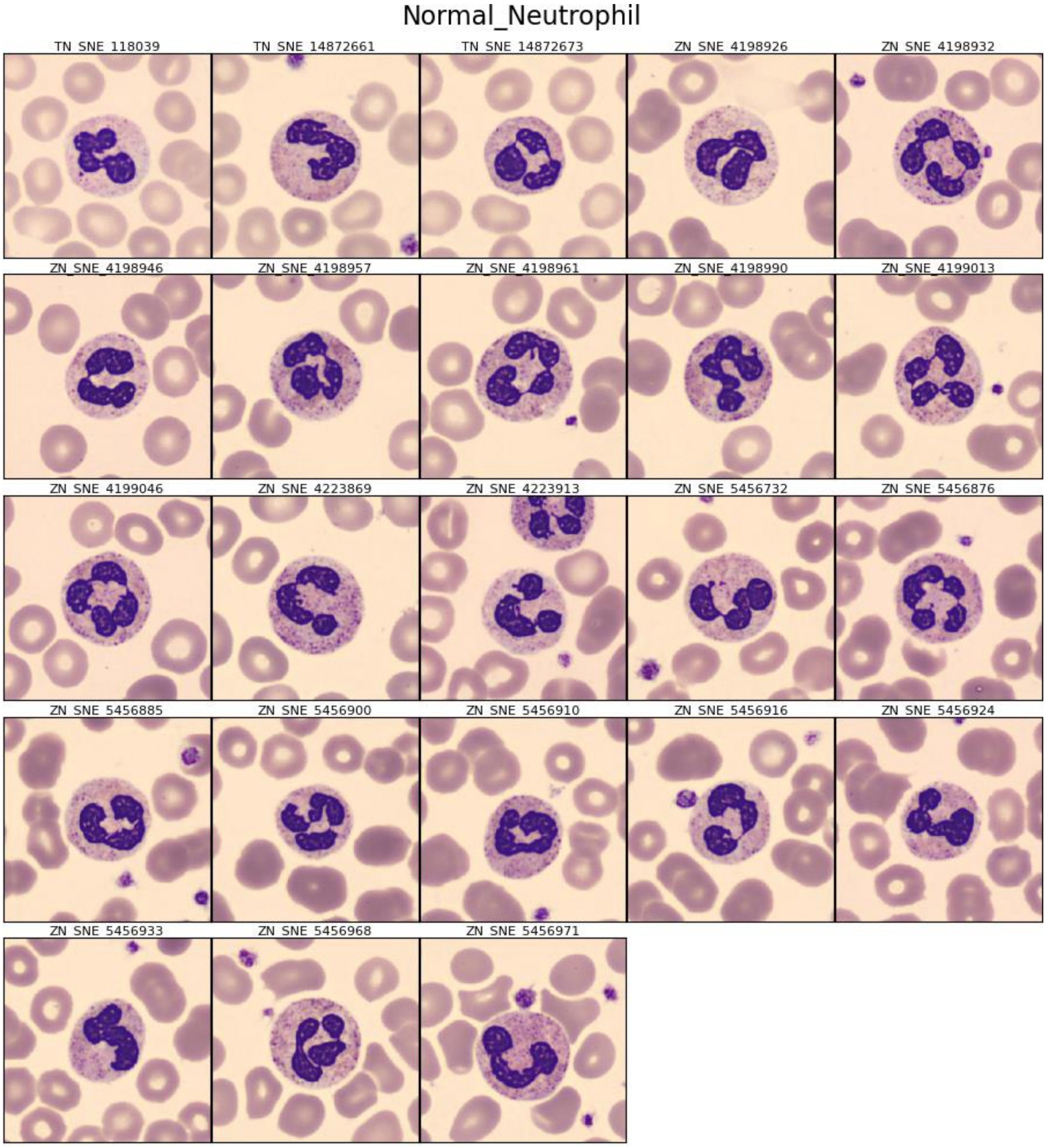
Image Name	Dataset Class	DisplasiaNet		Cytoplasm		Chromatin		Lobe		Total score	Pathologist Diagnosis
		Score	Prediction	State	Score	State	Score	State	Score		
WD_BNE_2426191	0	0.9889	1	Transparent	3	Heterogenic	3	Hyposegmented	1	7	DYSPLASTIC
WD_SNE_2426196	0	0.9997	1	Transparent	1	Heterogenic	1	Hyposegmented	1	3	DYSPLASTIC
WD_SNE_2683958	0	1.0000	1	Transparent	2	Heterogenic	3	Hyposegmented	1	6	DYSPLASTIC
WD_SNE_2288170	0	1.0000	1	Granulated	-1	Homogenic	-3	Hipersegmented	1	-3	NORMAL
WD_SNE_2426212	0	1.0000	1	Granulated	-1	Heterogenic	1	Normal	0	0	NORMAL

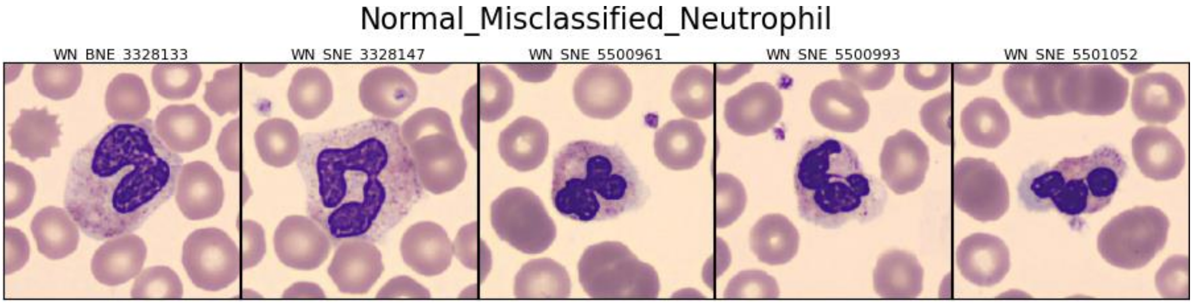
Image Name	Dataset Class	DisplasiaNet		Cytoplasm		Chromatin		Lobe		Total score	Pathologist Diagnosis
		Score	Prediction	State	Score	State	Score	State	Score		
TN_SNE_118039	1	1.0000	1	Granulated	-1	Homo.	-1	Normal	0	-2	NORMAL
TN_SNE_14872661	1	0.9999	1	Granulated	-1	Homo.	1	Normal	0	0	NORMAL
TN_SNE_14872673	1	1.0000	1	Granulated	-1	Homo.	1	normal	0	0	NORMAL
ZN_SNE_4198926	1	1.0000	1	Granulated	-1	Homo.	-2	Normal	0	-3	NORMAL
ZN_SNE_4198932	1	1.0000	1	Granulated	-2	Homo.	1	Normal	0	-1	NORMAL
ZN_SNE_4198946	1	1.0000	1	Granulated	-2	Homo.	-1	Normal	0	-3	NORMAL
ZN_SNE_4198957	1	1.0000	1	Granulated	-2	Homo.	-2	Normal	0	-4	NORMAL
ZN_SNE_4198961	1	1.0000	1	Granulated	-2	Homo.	1	normal	0	-1	NORMAL
ZN_SNE_4198990	1	1.0000	1	Granulated	-3	Homo.	2	normal	0	-1	NORMAL
ZN_SNE_4199013	1	1.0000	1	Granulated	-2	Homo.	-3	Normal	0	-5	NORMAL
ZN_SNE_4199046	1	1.0000	1	Granulated	-2	Homo.	-2	Normal	0	-4	NORMAL
ZN_SNE_4223869	1	1.0000	1	Granulated	-3	Homo.	-2	Normal	0	-5	NORMAL
ZN_SNE_4223913	1	0.9916	1	Granulated	-2	Homo.	-1	normal	0	-3	NORMAL
ZN_SNE_5456732	1	1.0000	1	Granulated	-2	Homo.	-1	normal	0	-3	NORMAL
ZN_SNE_5456876	1	1.0000	1	Granulated	-2	Homo.	-1	normal	0	-3	NORMAL
ZN_SNE_5456885	1	1.0000	1	Granulated	-2	Homo.	-1	normal	0	-3	NORMAL
ZN_SNE_5456900	1	0.9998	1	Granulated	-1	Homo.	-2	Normal	0	-3	NORMAL
ZN_SNE_5456910	1	1.0000	1	Granulated	-2	Homo.	-2	Normal	0	-4	NORMAL
ZN_SNE_5456916	1	1.0000	1	Granulated	-2	Homo.	1	normal	0	-1	NORMAL
ZN_SNE_5456924	1	1.0000	1	Granulated	-2	Homo.	-1	normal	0	-3	NORMAL
ZN_SNE_5456933	1	0.9998	1	Granulated	-2	Homo.	-1	normal	0	-3	NORMAL
ZN_SNE_5456968	1	1.0000	1	Granulated	-2	Homo.	-2	Normal	0	-4	NORMAL
ZN_SNE_5456971	1	1.0000	1	Granulated	-3	Homo.	-3	Hyposegmented	1	-5	NORMAL

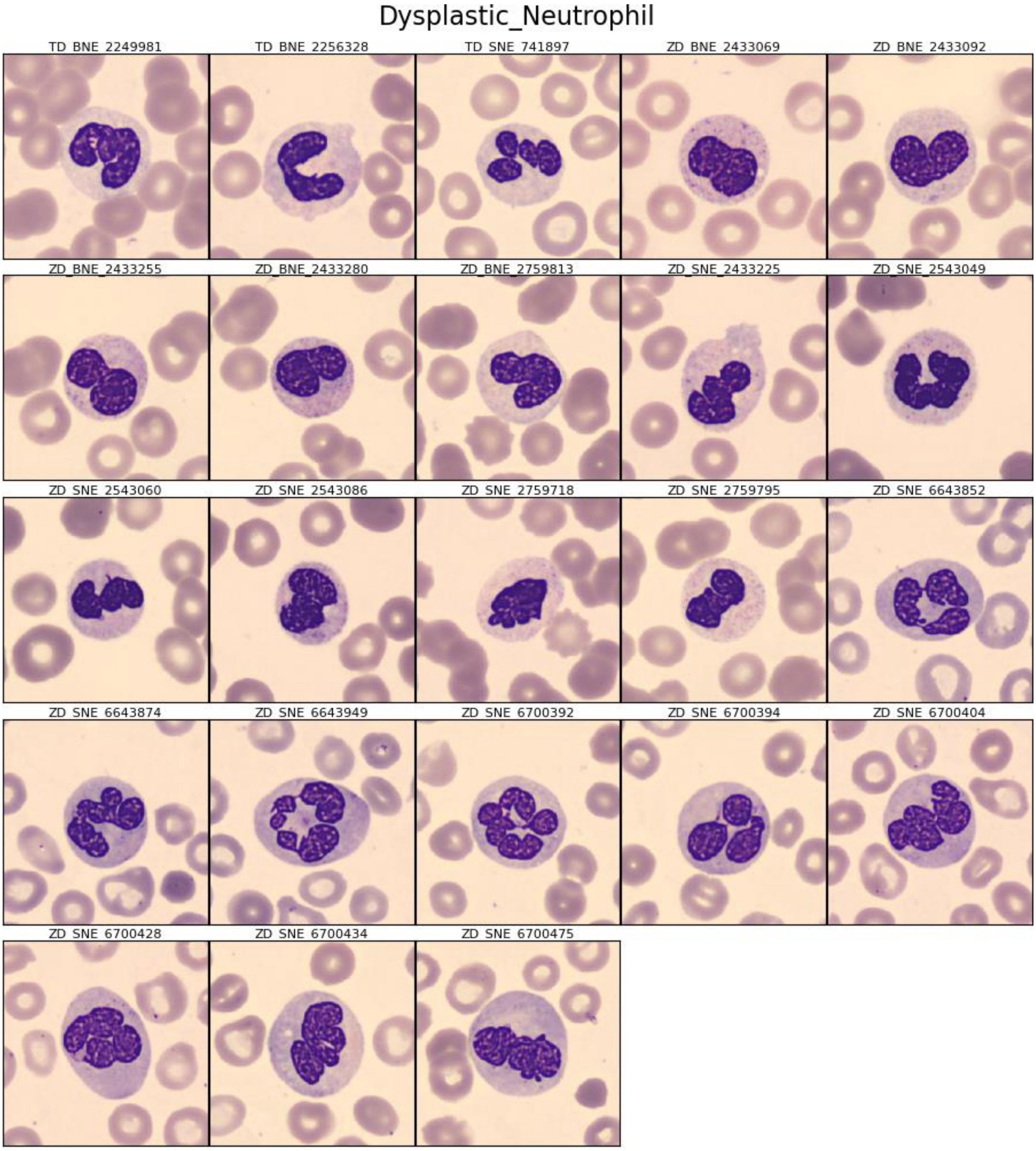
Image Name	Dataset Class	DisplasiaNet		Cytoplasm		Chromatin		Lobe		Total score	Pathologist Diagnosis
		Score	Prediction	State	Score	State	Score	State	Score		
WN_BNE_3328133	1	0.0126	0	Granulated	-1	Heterogenic	1	Hyposegmented	1	1	DYSPLASTIC
WN_SNE_3328147	1	0.0541	0	Granulated	-1	Heterogenic	1	Normal	0	0	NORMAL
WN_SNE_5500961	1	0.5496	0	Granulated	-2	Homogenic	-1	normal	0	-3	NORMAL
WN_SNE_5500993	1	0.0014	0	Granulated	-1	Homogenic	-2	Normal	0	-3	NORMAL
WN_SNE_5501052	1	0.0001	0	Granulated	-2	Homogenic	-1	Normal	1	-2	NORMAL

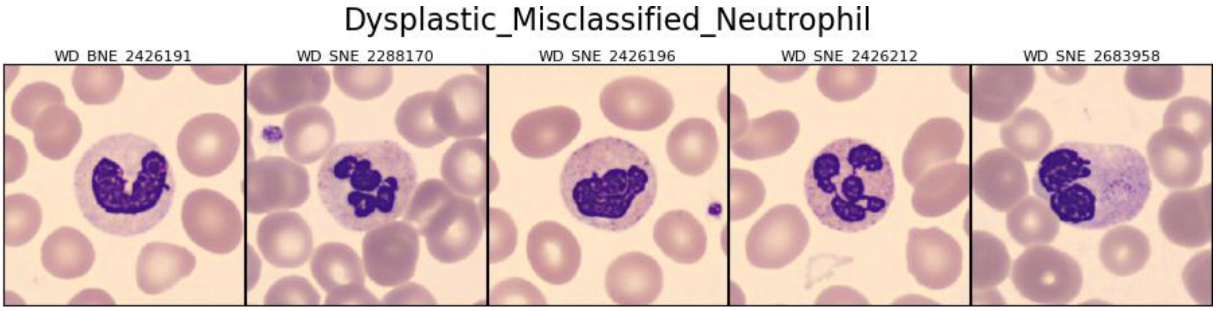
Annex C. Image Collection

C1. Study Dataset Images

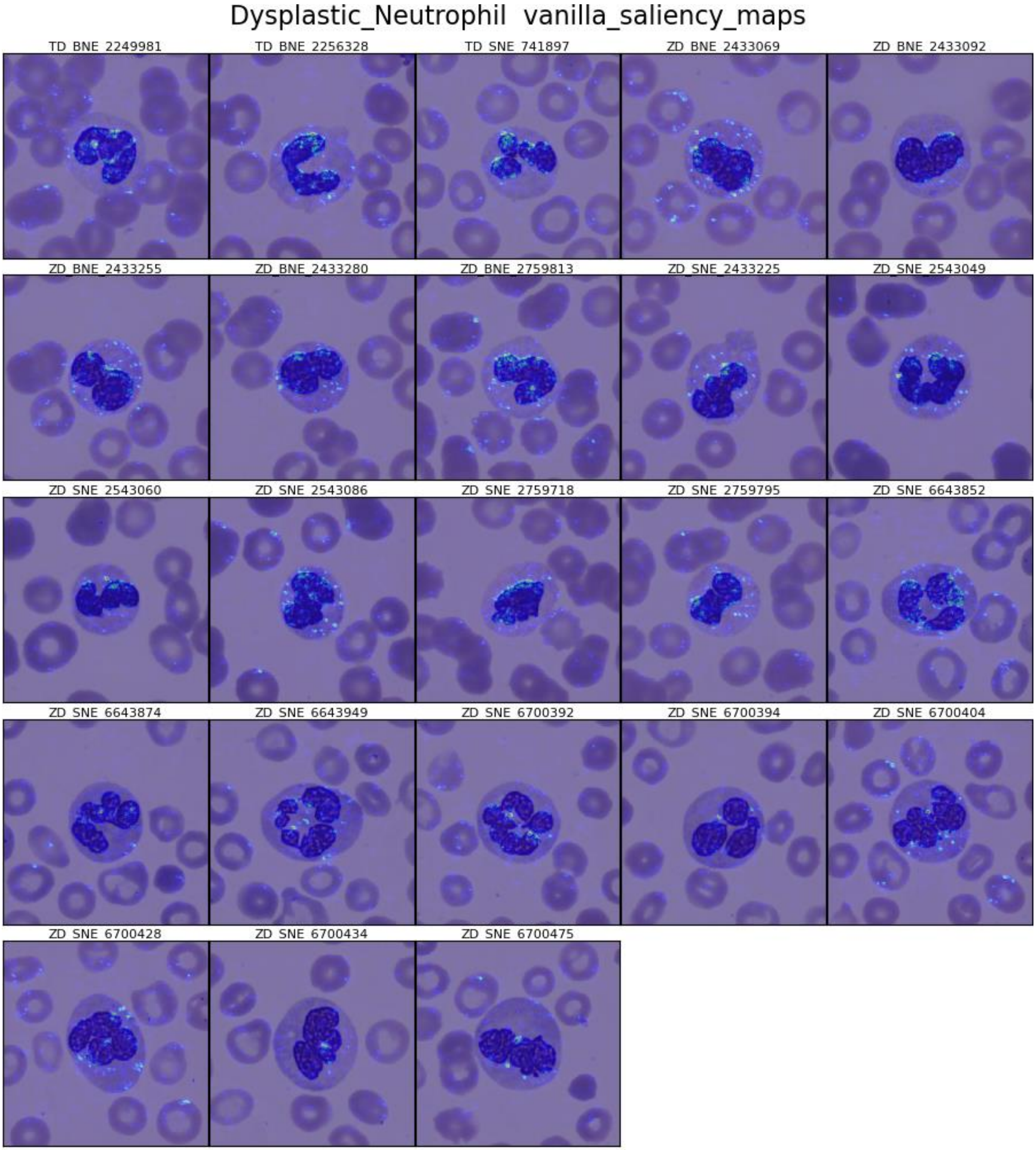


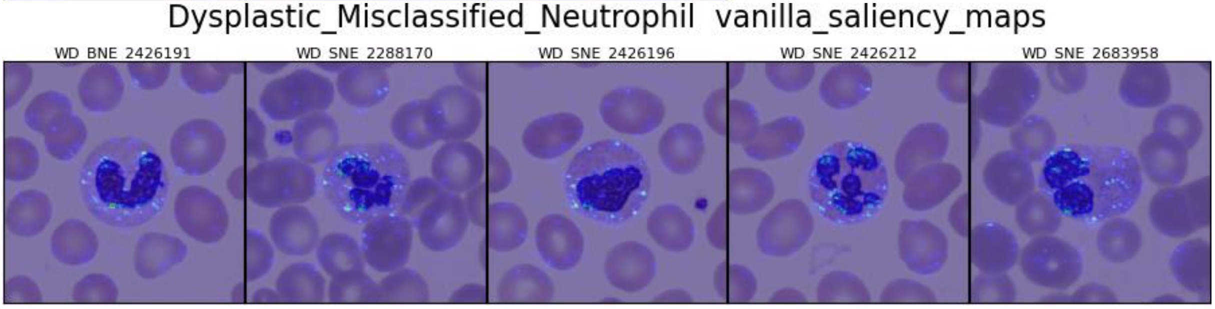


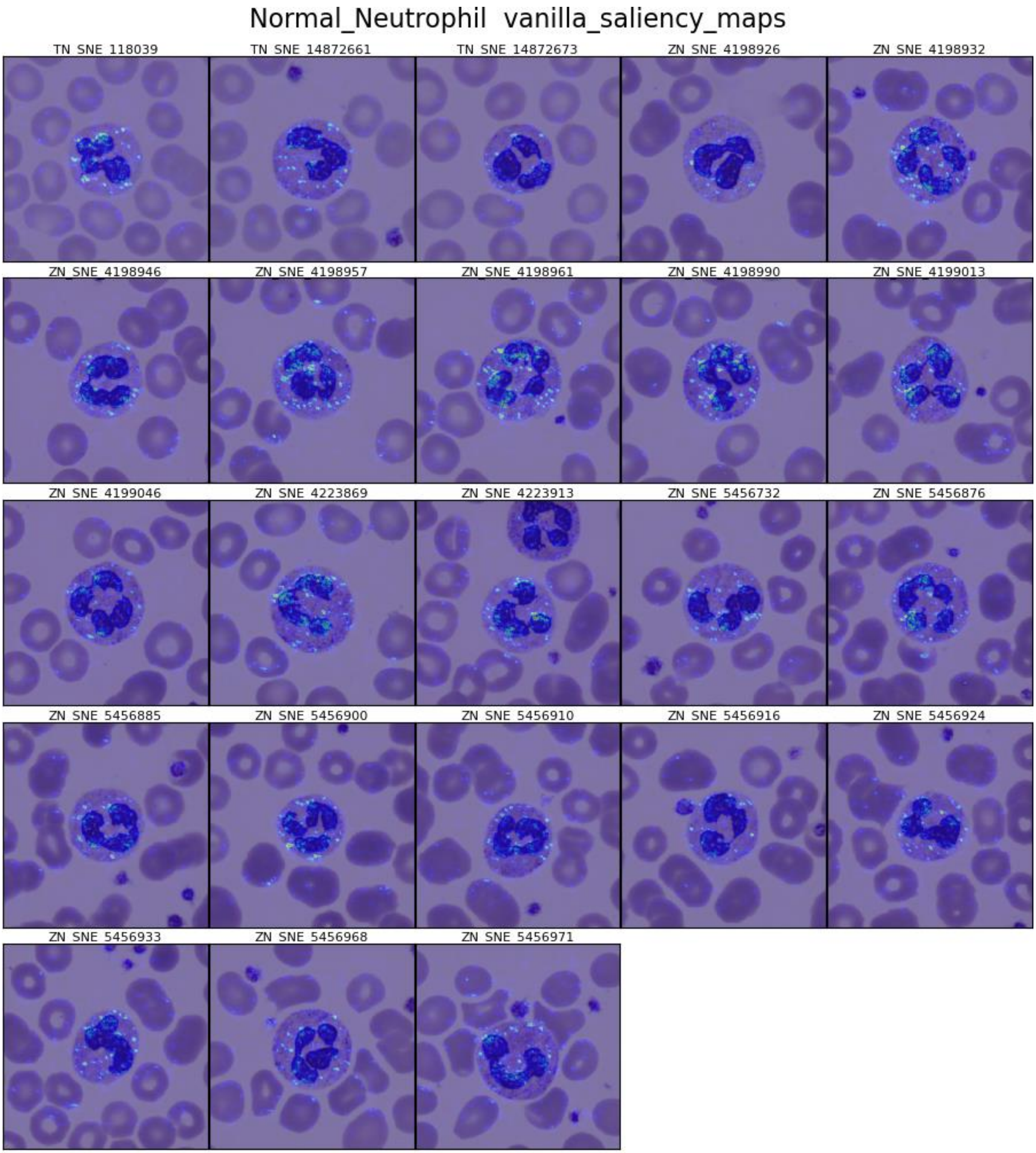

Dysplastic_Neutrophil
TD BNE 2249981
TD BNE 2256328
TD SNE 741897
ZD BNE 2433069
ZD BNE 2433092
ZD_BNE_2433255
ZD_BNE_2433280
ZD_BNE_2759813
ZD_SNE_2433225
ZD_SNE_2543049
ZD SNE 2543060
ZD SNE 2543086
ZD SNE 2759718
ZD SNE 2759795
ZD SNE 6643852
ZD SNE 6643874
ZD SNE 6643949
ZD SNE 6700392
ZD SNE 6700394
ZD SNE 6700404
ZD SNE 6700428
ZD SNE 6700434
ZD SNE 6700475

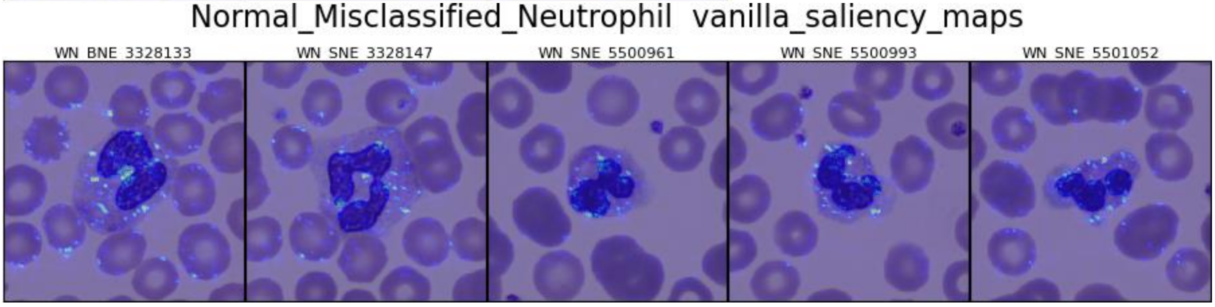

Dysplastic_Misclassified_Neutrophil
WD BNE 2426191
WD SNE 2288170
WD SNE 2426196
WD SNE 2426212
WD SNE 2683958

C2. Saliency Vanilla Maps

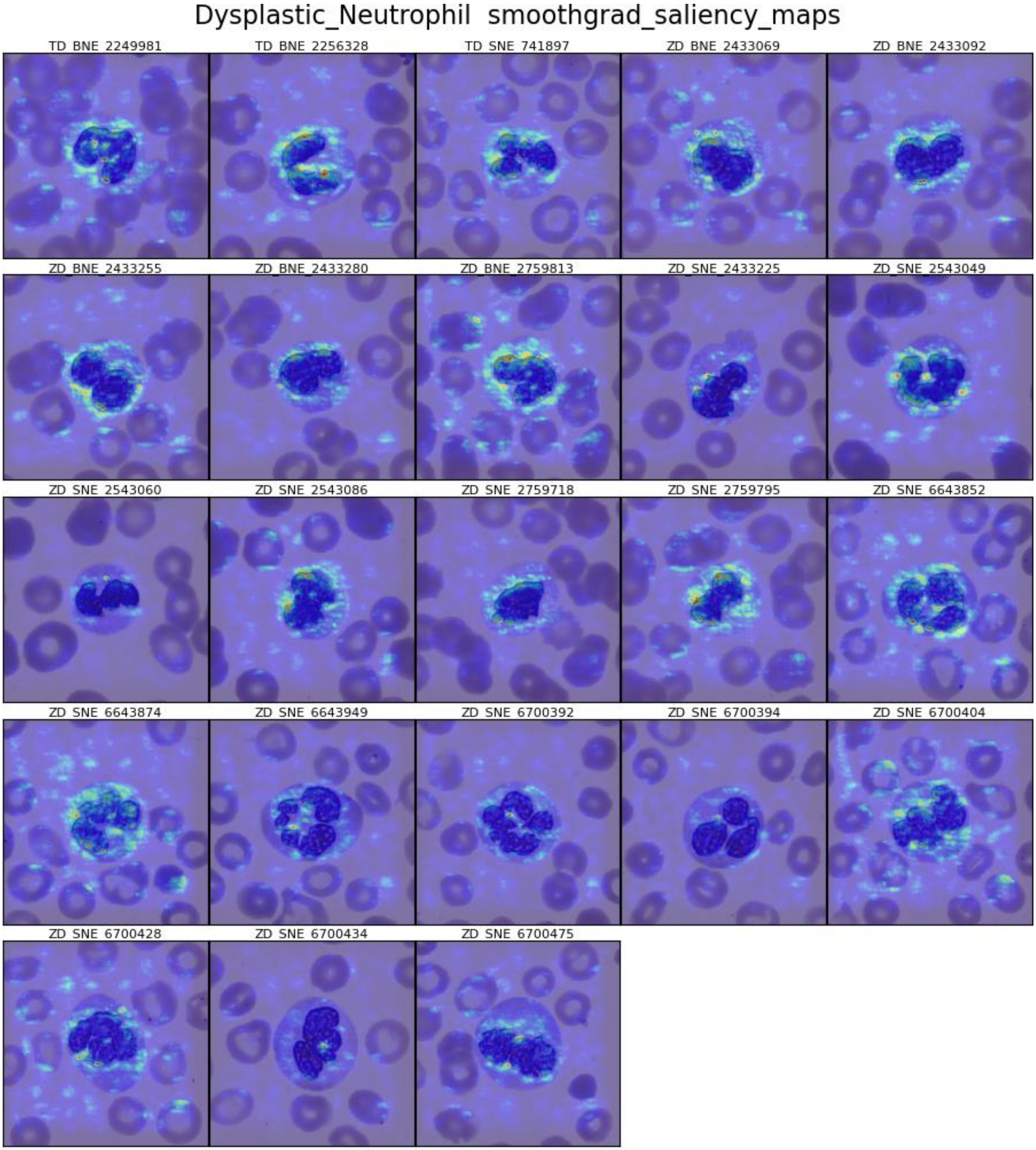


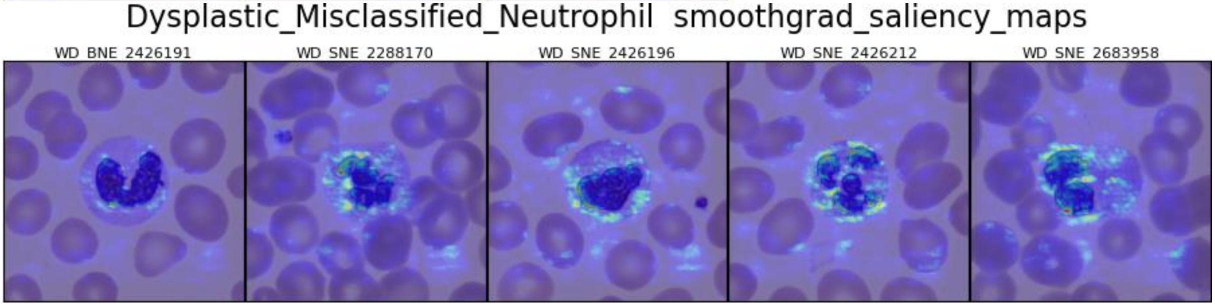


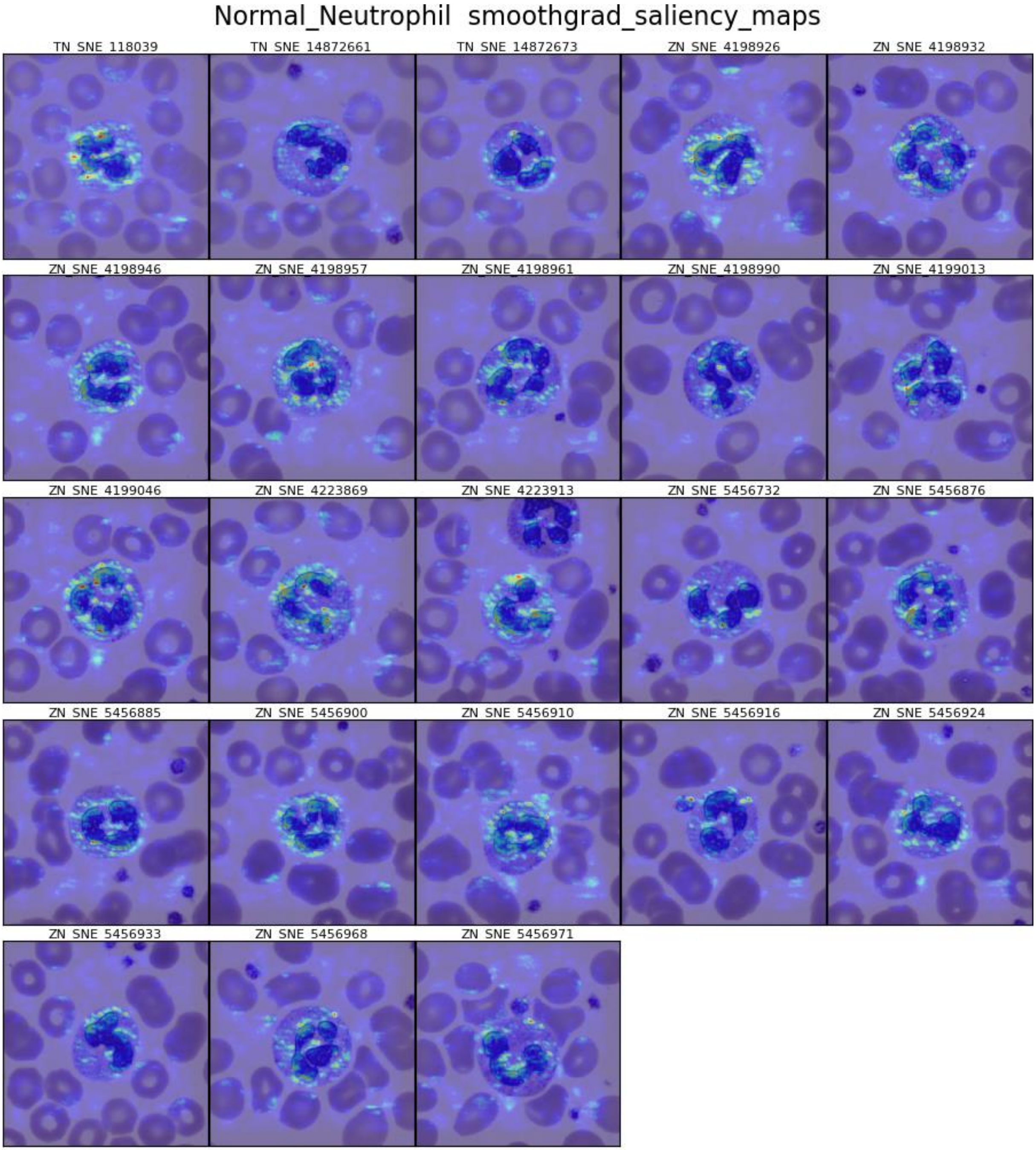

Normal_Neutrophil vanilla_saliency_maps
TN SNE 118039
TN SNE 14872661
TN SNE 14872673
ZN SNE 4198926
ZN SNE 4198932
ZN_SNE_4198946
ZN_SNE_4198957
ZN_SNE_4198961
ZN_SNE_4198990
ZN_SNE_4199013
ZN SNE 4199046
ZN SNE 4223869
ZN SNE 4223913
ZN SNE 5456732
ZN SNE 5456876
ZN SNE 5456885
ZN SNE 5456900
ZN SNE 5456910
ZN SNE 5456916
ZN SNE 5456924
ZN SNE 5456933
ZN SNE 5456968
ZN SNE 5456971

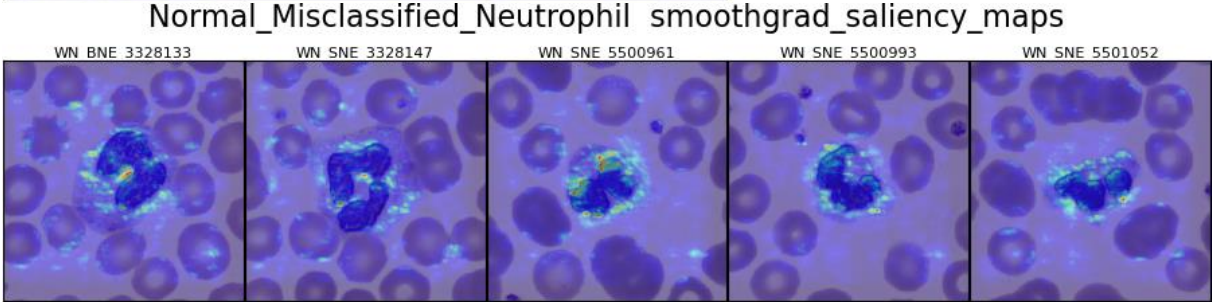

Normal_Misclassified_Neutrophil vanilla_saliency_maps
WN BNE 3328133
WN SNE 3328147
WN SNE 5500961
WN SNE 5500993
WN SNE 5501052

C3. Smoothgrad Saliency Maps

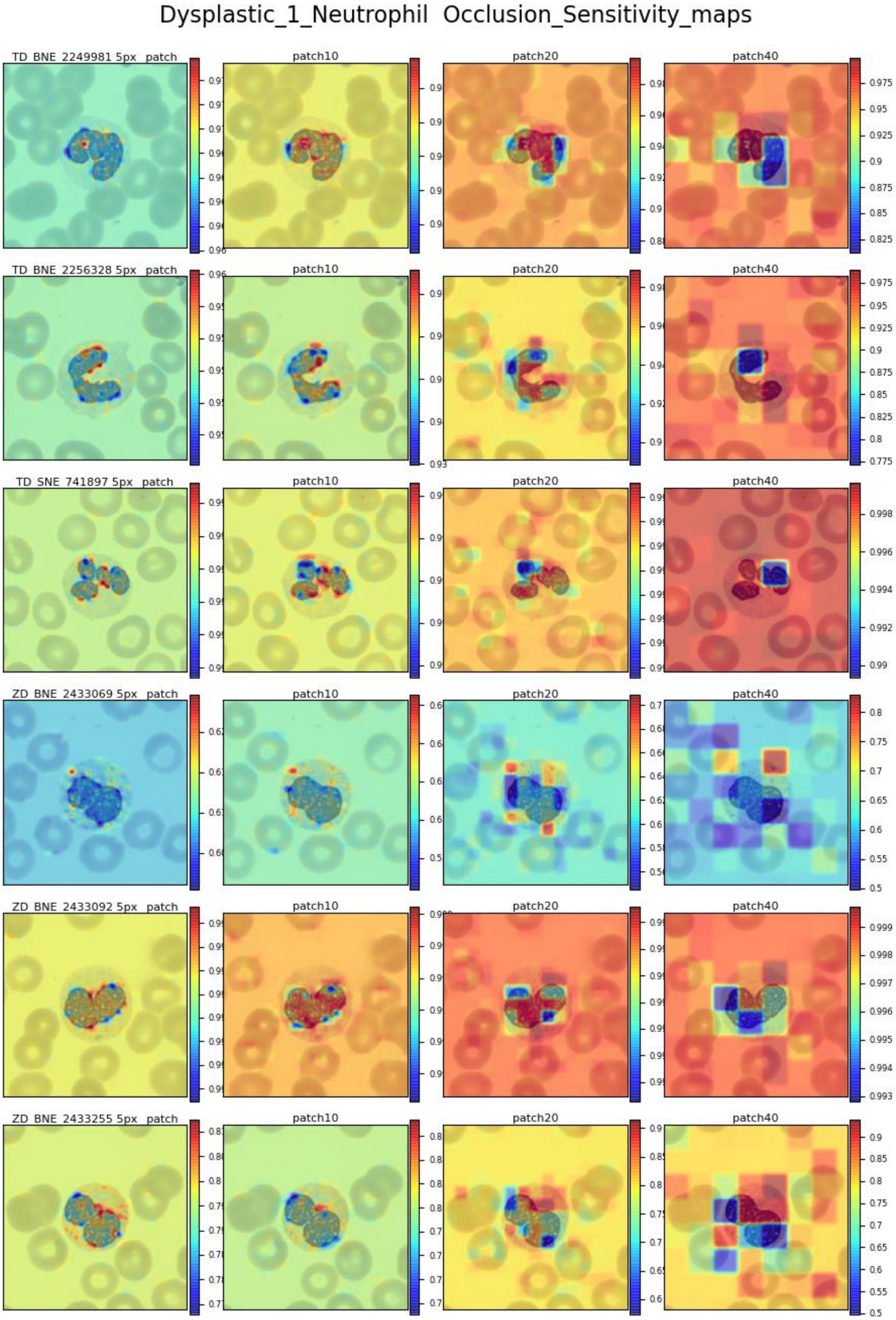


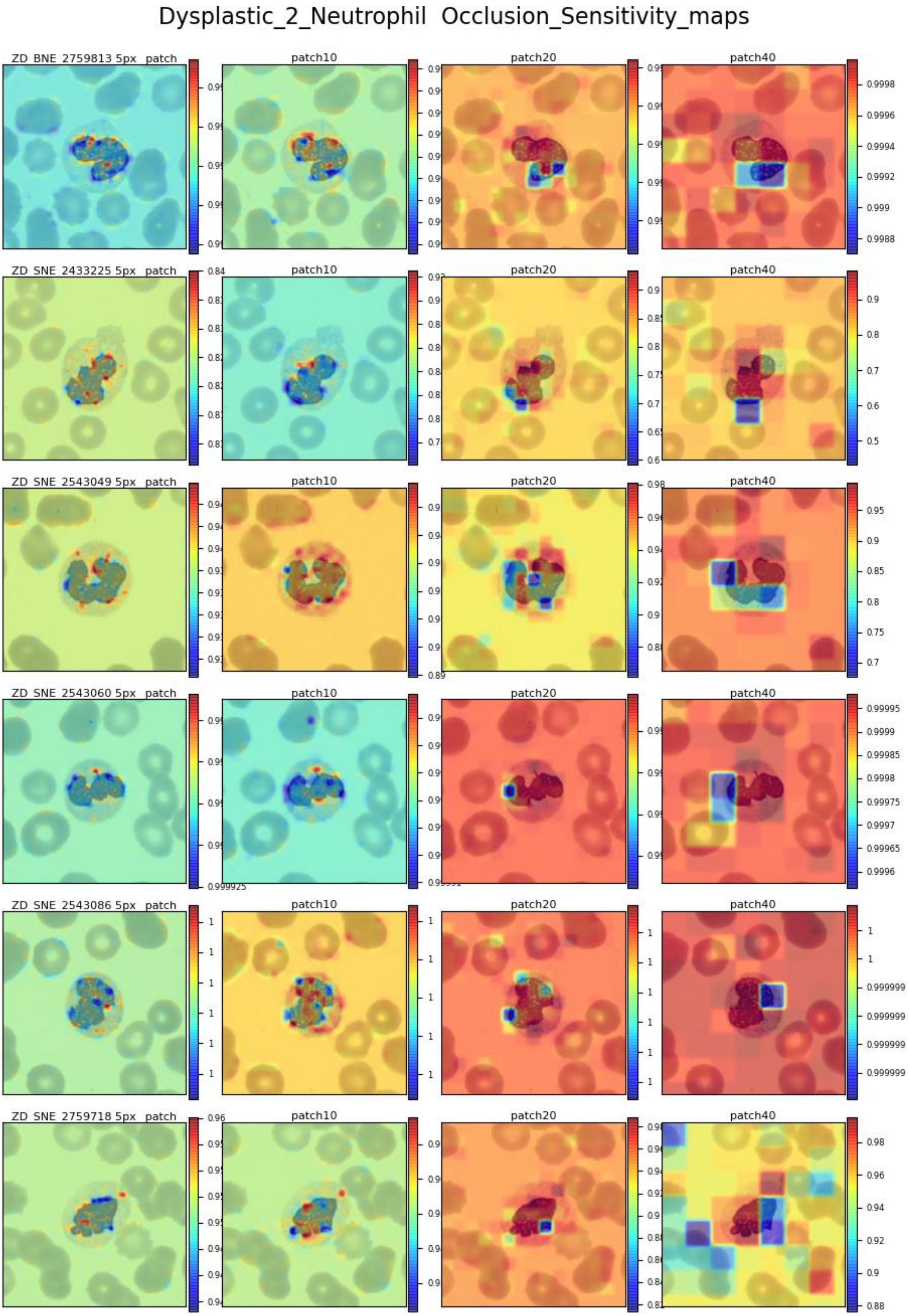



Normal_Neutrophil smoothgrad_saliency_maps
TN SNE 118039
TN SNE 14872661
TN SNE 14872673
ZN SNE 4198926
ZN SNE 4198932
ZN_SNE_4198946
ZN_SNE_4198957
ZN_SNE_4198961
ZN_SNE_4198990
ZN_SNE_4199013
ZN SNE 4199046
ZN SNE 4223869
ZN SNE 4223913
ZN SNE 5456732
ZN SNE 5456876
ZN SNE 5456885
ZN SNE 5456900
ZN SNE 5456910
ZN SNE 5456916
ZN SNE 5456924
ZN SNE 5456933
ZN SNE 5456968
ZN SNE 5456971

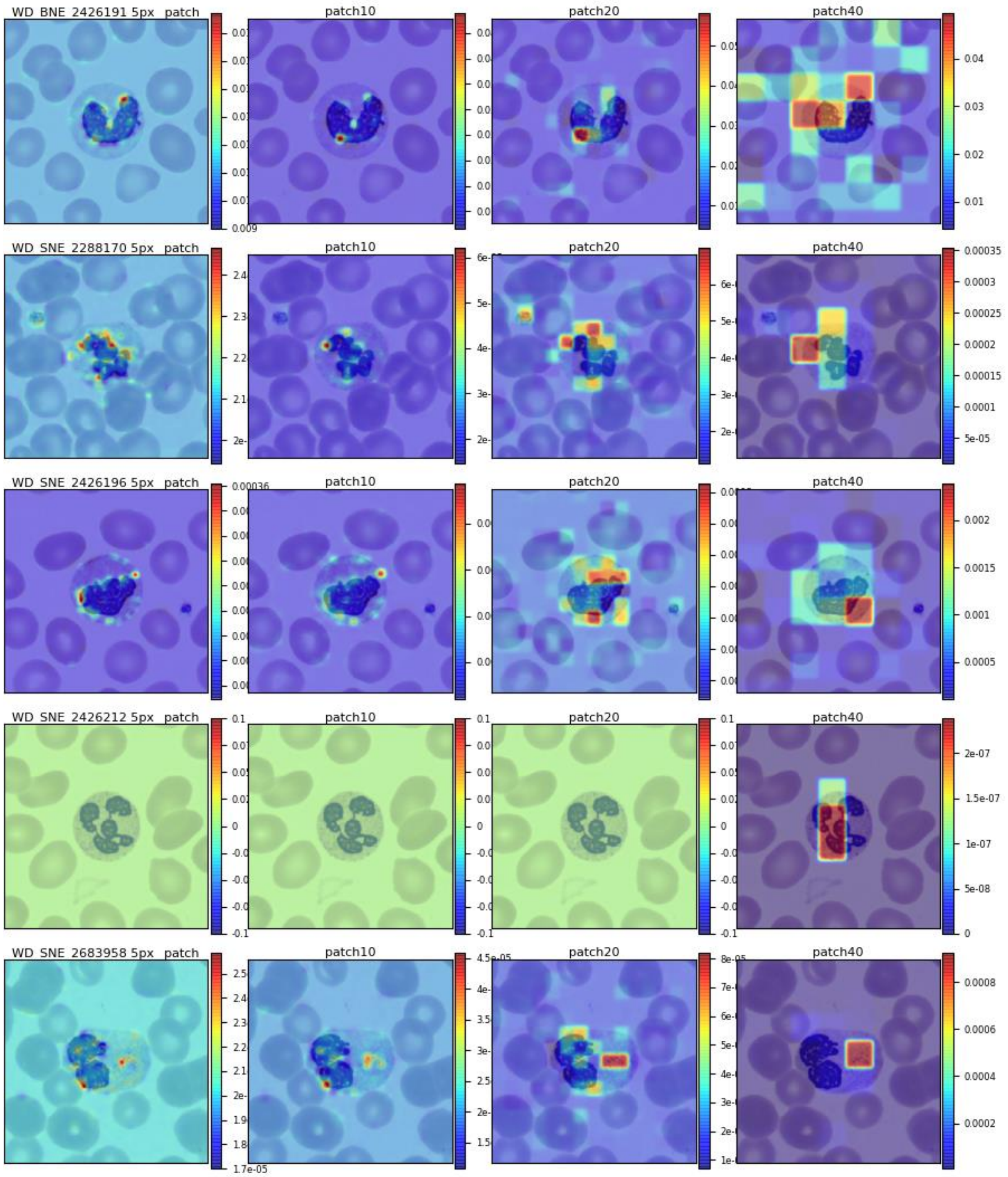

Normal_Misclassified_Neutrophil smoothgrad_saliency_maps
WN BNE 3328133
WN SNE 3328147
WN SNE 5500961
WN SNE 5500993
WN SNE 5501052

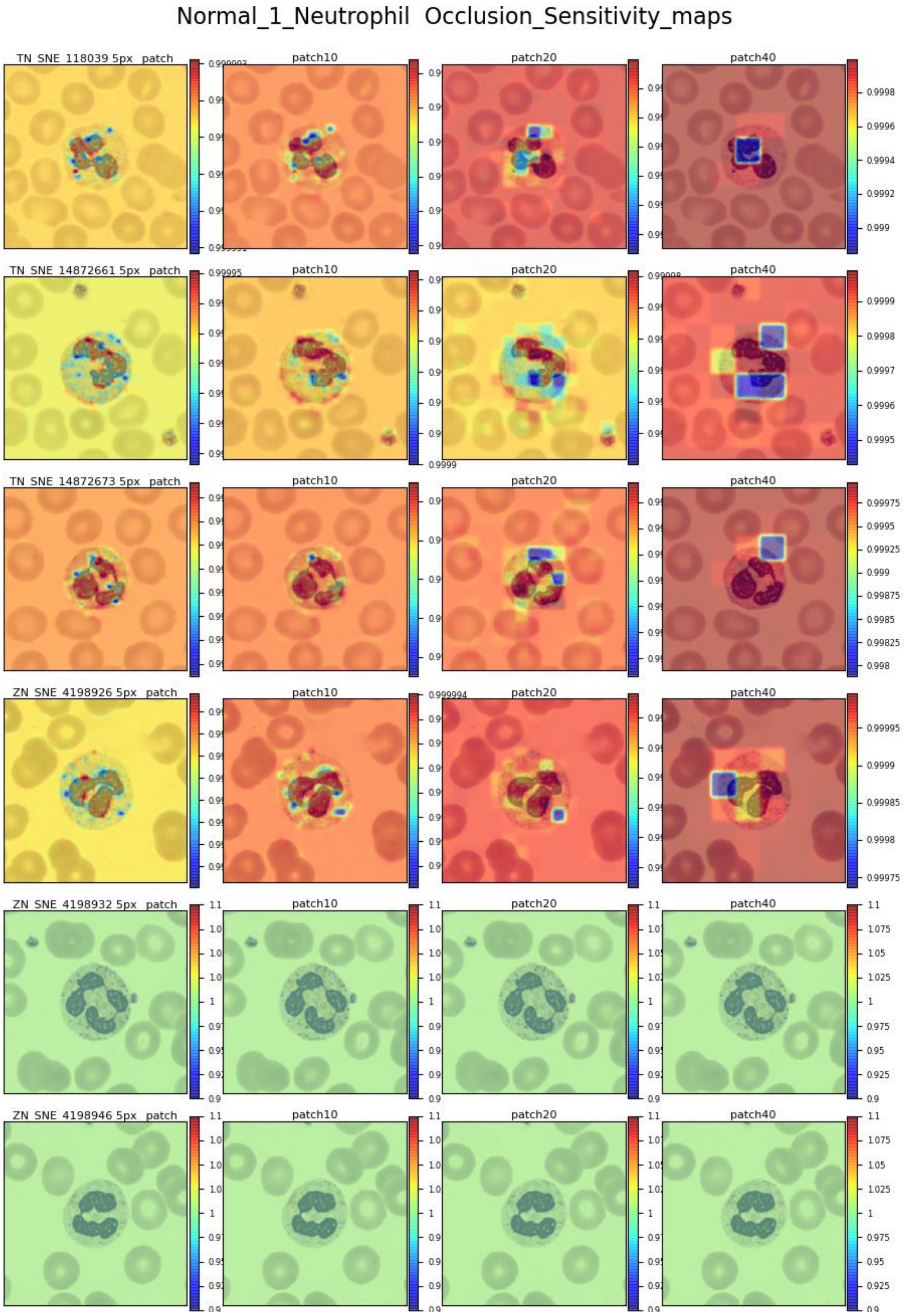
C4. Occlusion sensitivity Maps

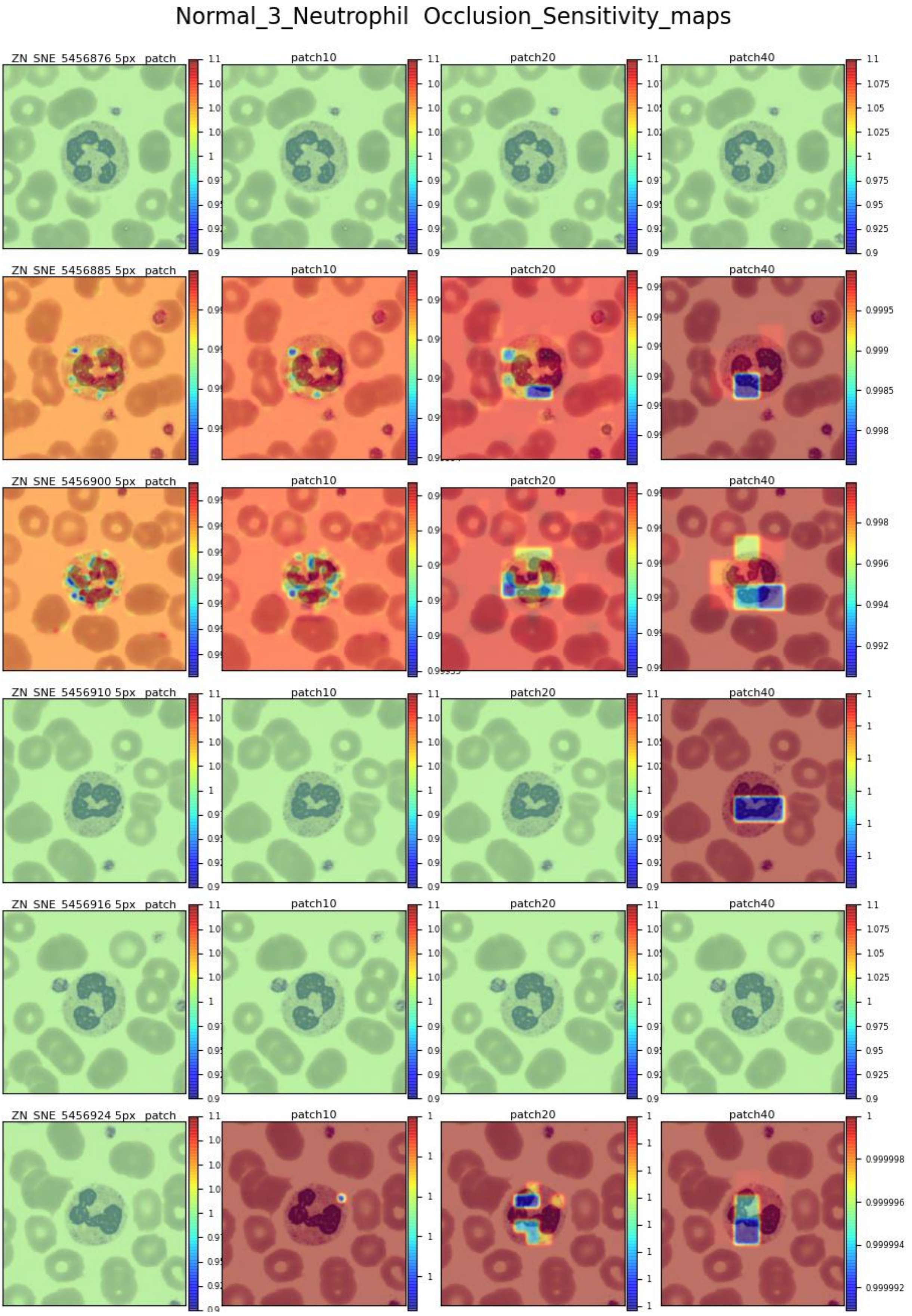


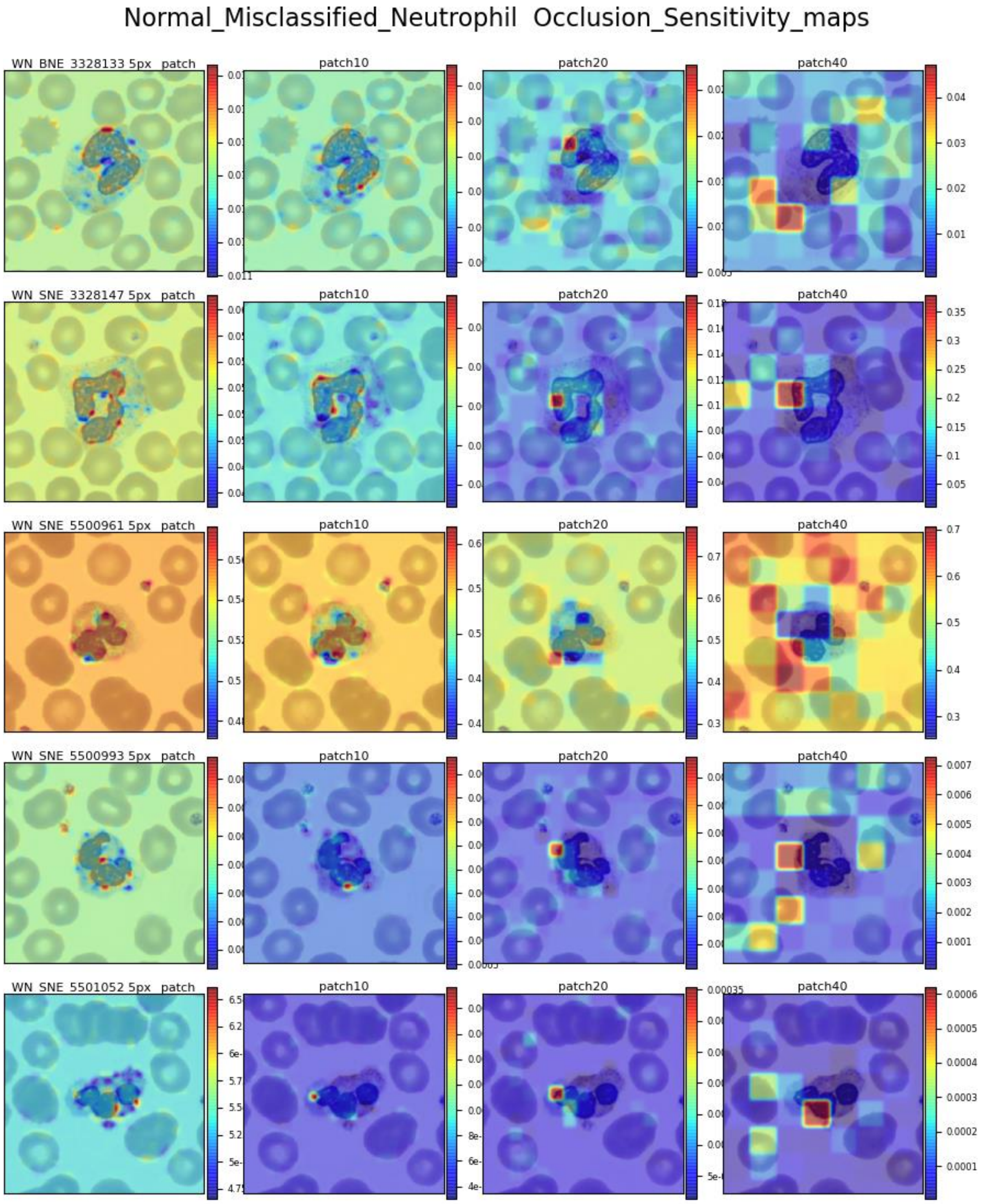

Dysplastic_2_Neutrophil Occlusion_Sensitivity_maps
ZD BNE 2759813 5px patch
ZD SNE 2433225 5px patch
ZD SNE 2543049 5px patch
ZD SNE 2543060 5px patch
ZD SNE 2543086 5px patch
ZD SNE 2759718 5px patch
patch10
patch20
patch40
0.9998
0.9996
0.9994
0.9992
0.999
0.9988
0.84
0.9
0.8
0.7
0.6
0.5
0.98
0.95
0.85
0.75
0.89
0.99995
0.9999
0.99985
0.9998
0.99975
0.9997
0.99965
0.9996
0.999925
1
0.999999
0.96
0.98
0.94
0.92
0.88

Dysplastic_Misclassified_Neutrophil Occlusion_Sensitivity_maps



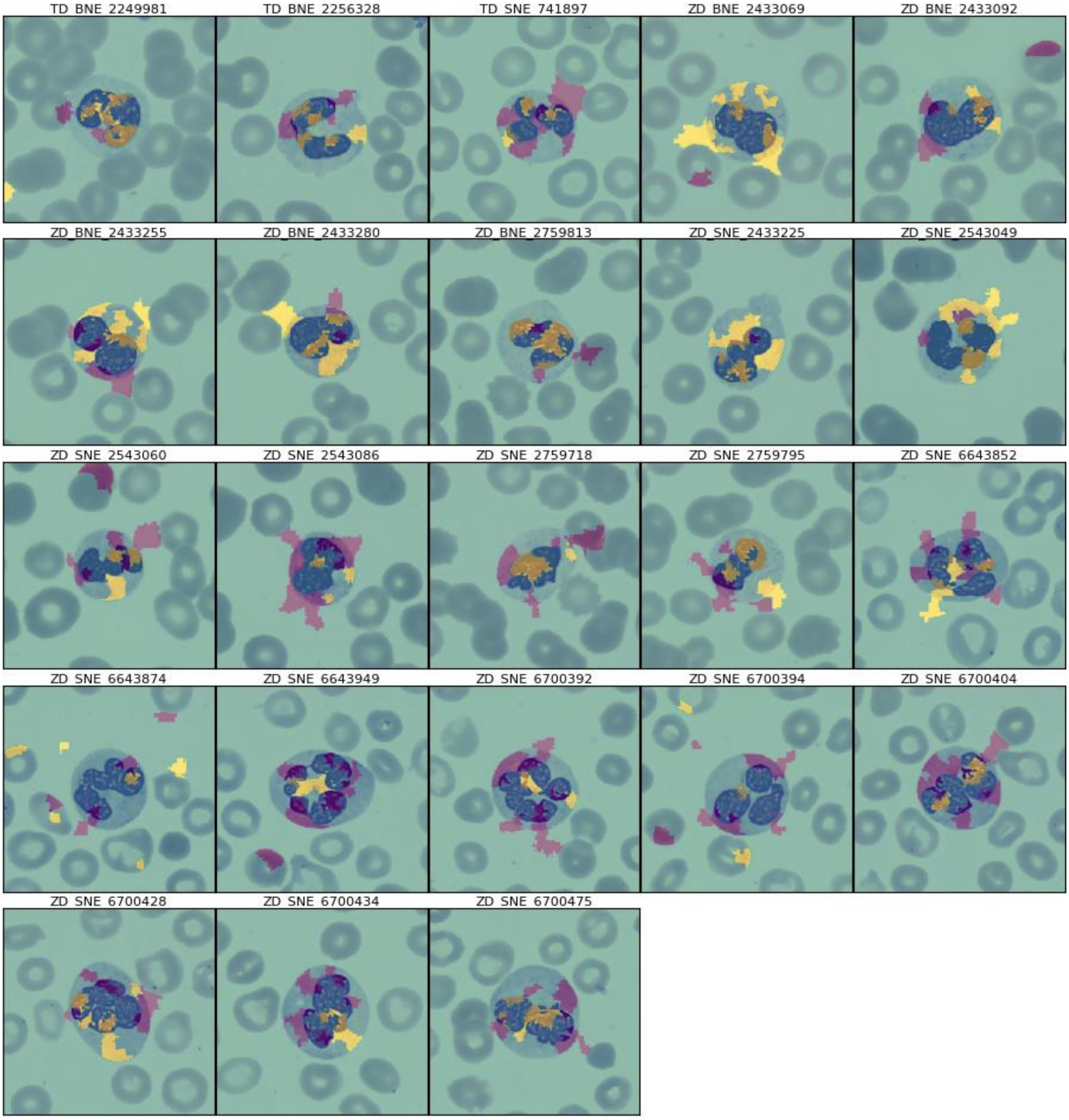

Normal_1_Neutrophil Occlusion_Sensitivity_maps
TN SNE 118039 5px patch
patch10
patch20
patch40
0.9998
0.9996
0.9994
0.9992
0.999
TN SNE 14872661 5px patch
patch10
patch20
patch40
0.9999
0.9998
0.9997
0.9996
0.9995
TN SNE 14872673 5px patch
patch10
patch20
patch40
0.99975
0.9995
0.99925
0.999
0.99875
0.9985
0.99825
0.998
ZN SNE 4198926 5px patch
patch10
patch20
patch40
0.99995
0.9999
0.99985
0.9998
0.99975
ZN SNE 4198932 5px patch
patch10
patch20
patch40
1.1
1.075
1.05
1.025
1
0.975
0.95
0.925
0.9
ZN SNE 4198946 5px patch
patch10
patch20
patch40
1.1
1.075
1.05
1.025
1
0.975
0.95
0.925
0.9


Normal_3_Neutrophil Occlusion_Sensitivity_maps
ZN_SNE_5456876 5px_patch
patch10
patch20
patch40
1.1
1.075
1.05
1.025
1
0.975
0.95
0.925
0.9
ZN_SNE_5456885 5px_patch
patch10
patch20
patch40
0.9995
0.999
0.9985
0.998
ZN_SNE_5456900 5px_patch
patch10
patch20
patch40
0.998
0.996
0.994
0.992
ZN_SNE_5456910 5px_patch
patch10
patch20
patch40
1.1
1
ZN_SNE_5456916 5px_patch
patch10
patch20
patch40
1.1
1.075
1.05
1.025
1
0.975
0.95
0.925
0.9
ZN_SNE_5456924 5px_patch
patch10
patch20
patch40
1
0.999998
0.999996
0.999994
0.999992


Normal_Misclassified_Neutrophil Occlusion_Sensitivity_maps
WN BNE 3328133 5px patch
patch10
patch20
patch40
0.04
0.03
0.02
0.01
WN SNE 3328147 5px patch
patch10
patch20
patch40
0.35
0.3
0.25
0.2
0.15
0.1
0.05
WN SNE 5500961 5px patch
patch10
patch20
patch40
0.7
0.6
0.5
0.4
0.3
WN SNE 5500993 5px patch
patch10
patch20
patch40
0.007
0.006
0.005
0.004
0.003
0.002
0.001
WN SNE 5501052 5px patch
patch10
patch20
patch40
0.0006
0.0005
0.0004
0.0003
0.0002
0.0001

C5. LIME

Dysplastic_Neutrophil lime_maps



TD BNE 2249981, TD BNE 2256328, TD SNE 741897, ZD BNE 2433069, ZD BNE 2433092

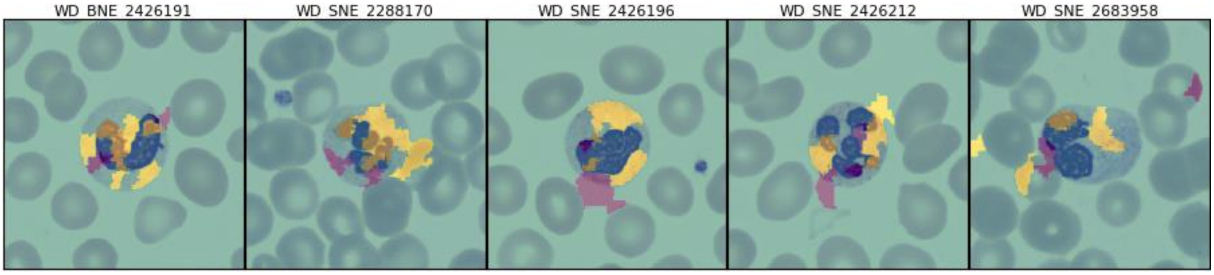
ZD_BNE_2433255, ZD_BNE_2433280, ZD_BNE_2759813, ZD_SNE_2433225, ZD_SNE_2543049

ZD SNE 2543060, ZD SNE 2543086, ZD SNE 2759718, ZD SNE 2759795, ZD SNE 6643852

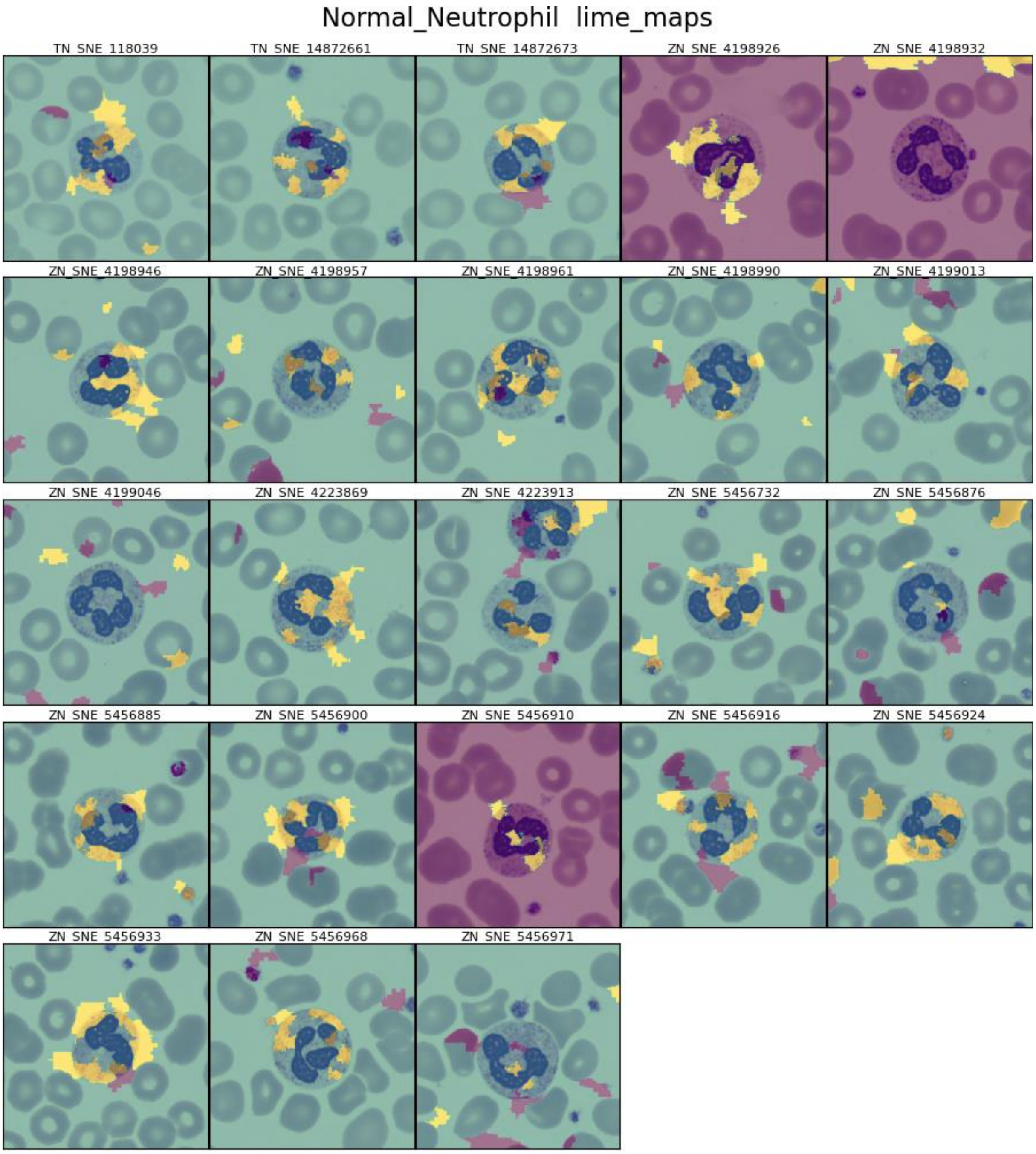
ZD SNE 6643874, ZD SNE 6643949, ZD SNE 6700392, ZD SNE 6700394, ZD SNE 6700404

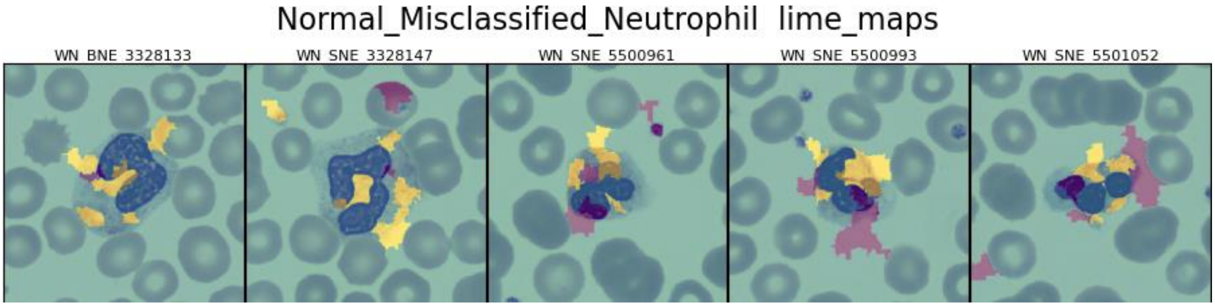
ZD SNE 6700428, ZD SNE 6700434, ZD SNE 6700475

Dysplastic_Misclassified_Neutrophil lime_maps

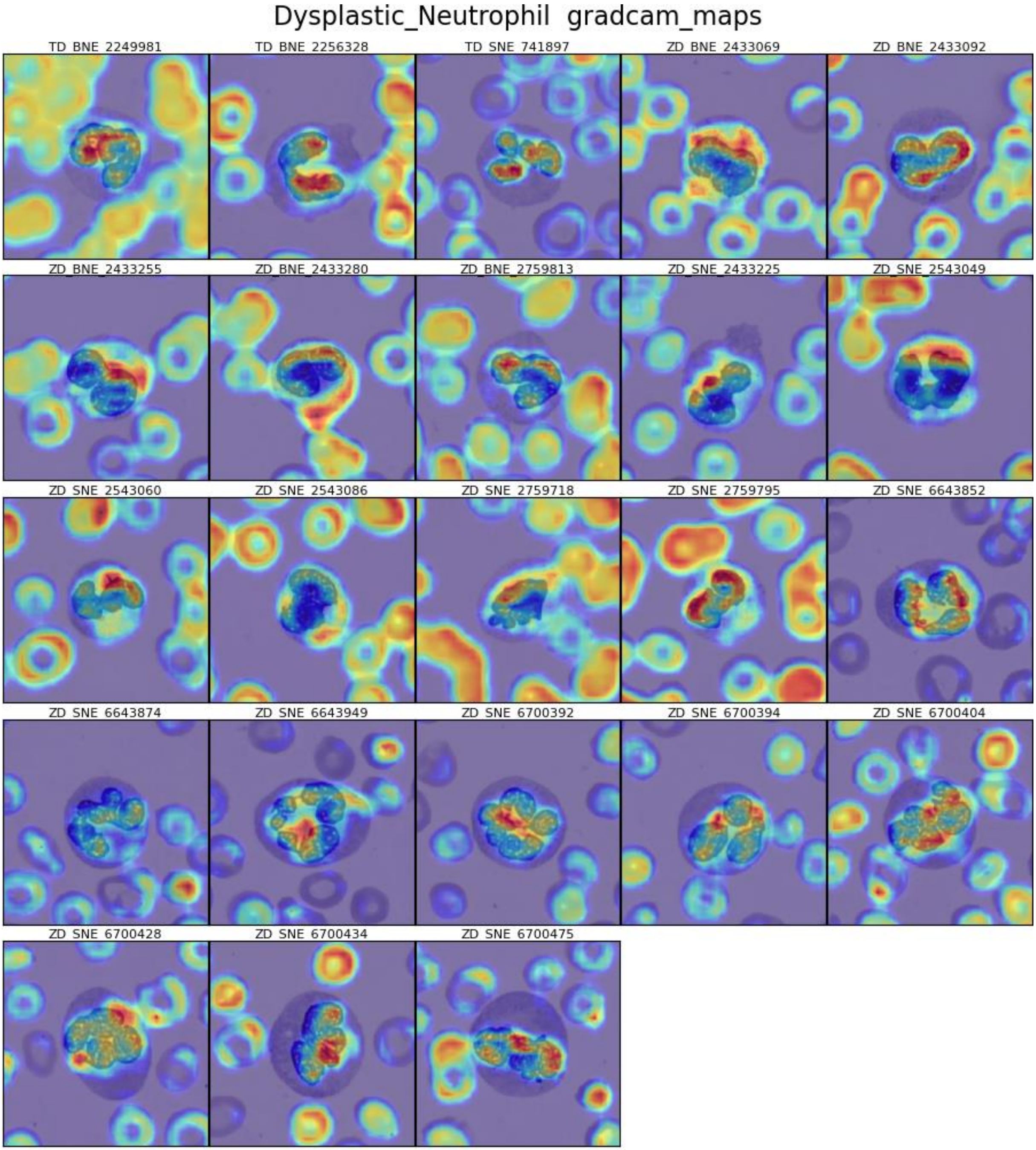


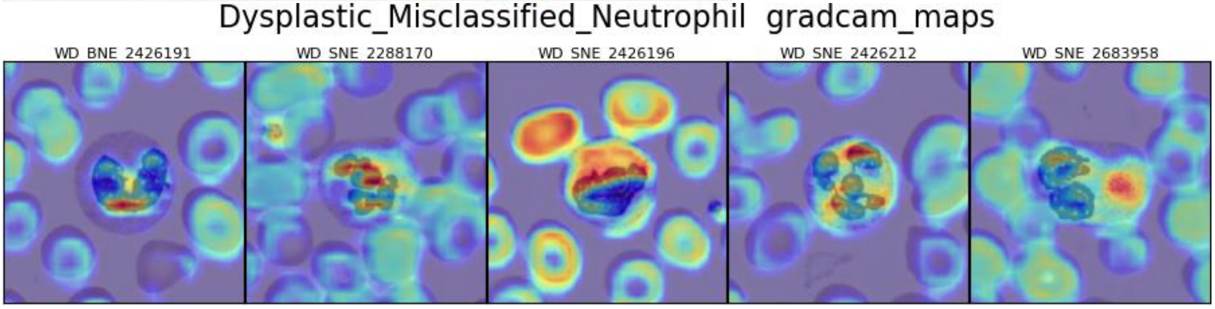
WD BNE 2426191, WD SNE 2288170, WD SNE 2426196, WD SNE 2426212, WD SNE 2683958

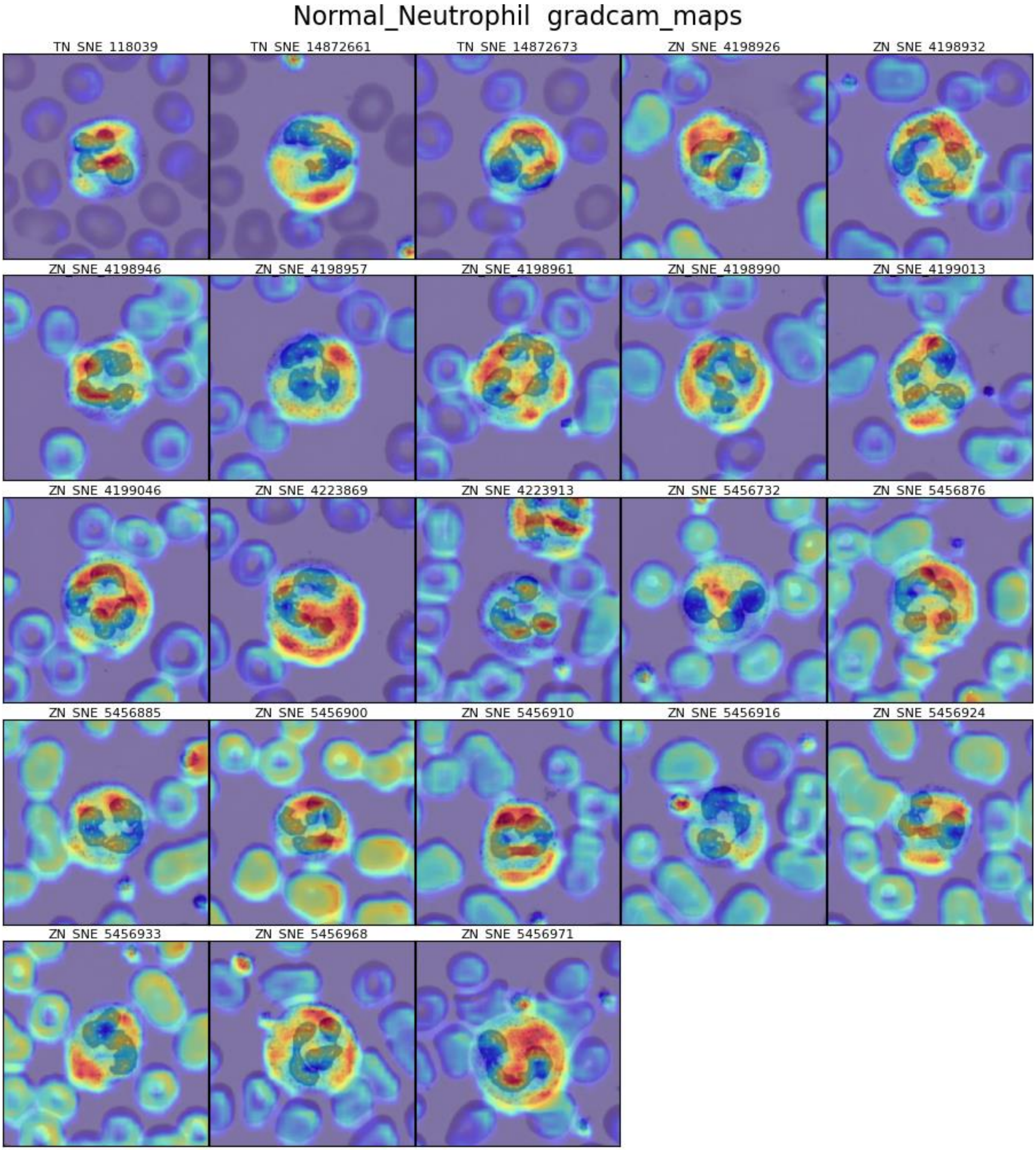


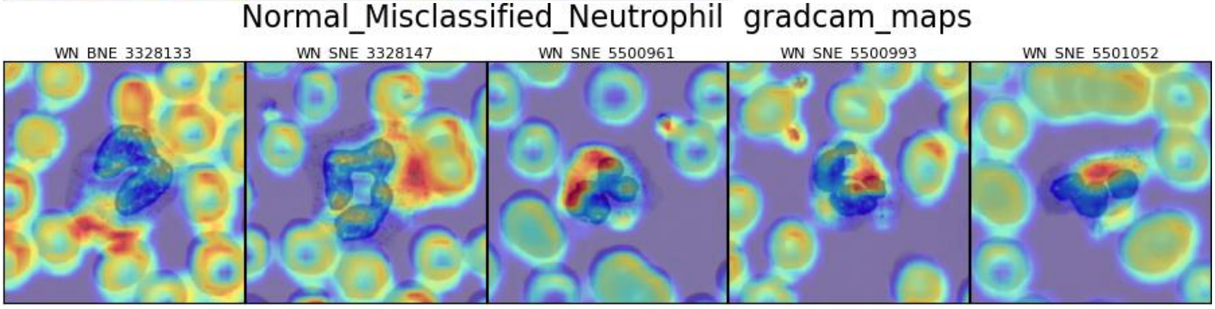


C6. Grad-CAM Maps

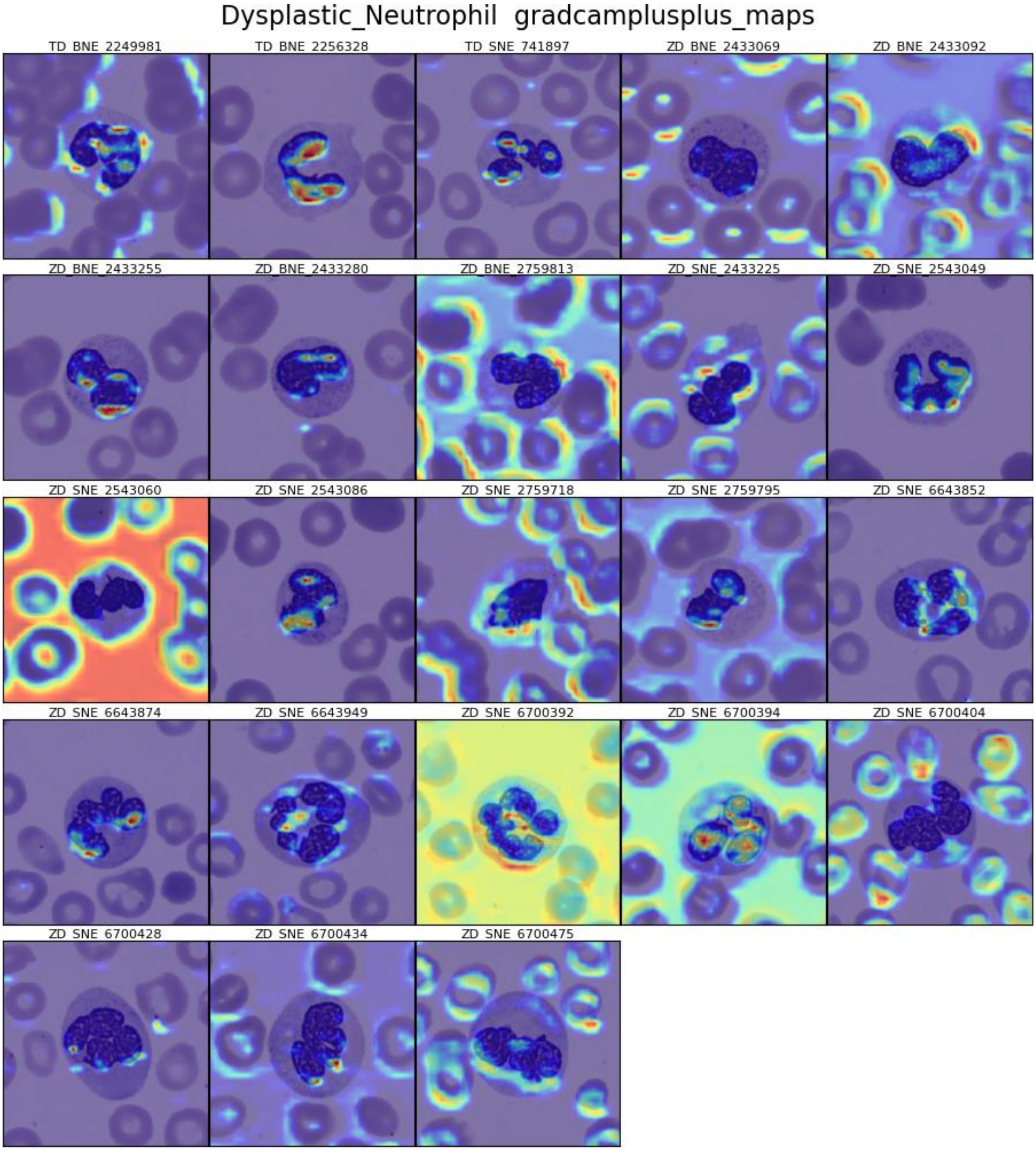

Dysplastic_Neutrophil gradcam_maps
TD BNE 2249981
TD BNE 2256328
TD SNE 741897
ZD BNE 2433069
ZD BNE 2433092
ZD_BNE_2433255
ZD_BNE_2433280
ZD_BNE_2759813
ZD_SNE_2433225
ZD_SNE_2543049
ZD SNE 2543060
ZD SNE 2543086
ZD SNE 2759718
ZD SNE 2759795
ZD SNE 6643852
ZD SNE 6643874
ZD SNE 6643949
ZD SNE 6700392
ZD SNE 6700394
ZD SNE 6700404
ZD SNE 6700428
ZD SNE 6700434
ZD SNE 6700475

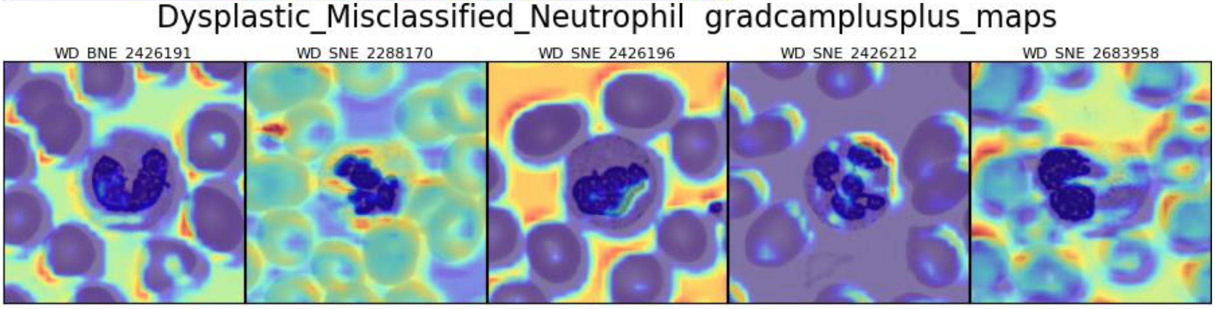

Dysplastic_Misclassified_Neutrophil gradcam_maps
WD BNE 2426191
WD SNE 2288170
WD SNE 2426196
WD SNE 2426212
WD SNE 2683958

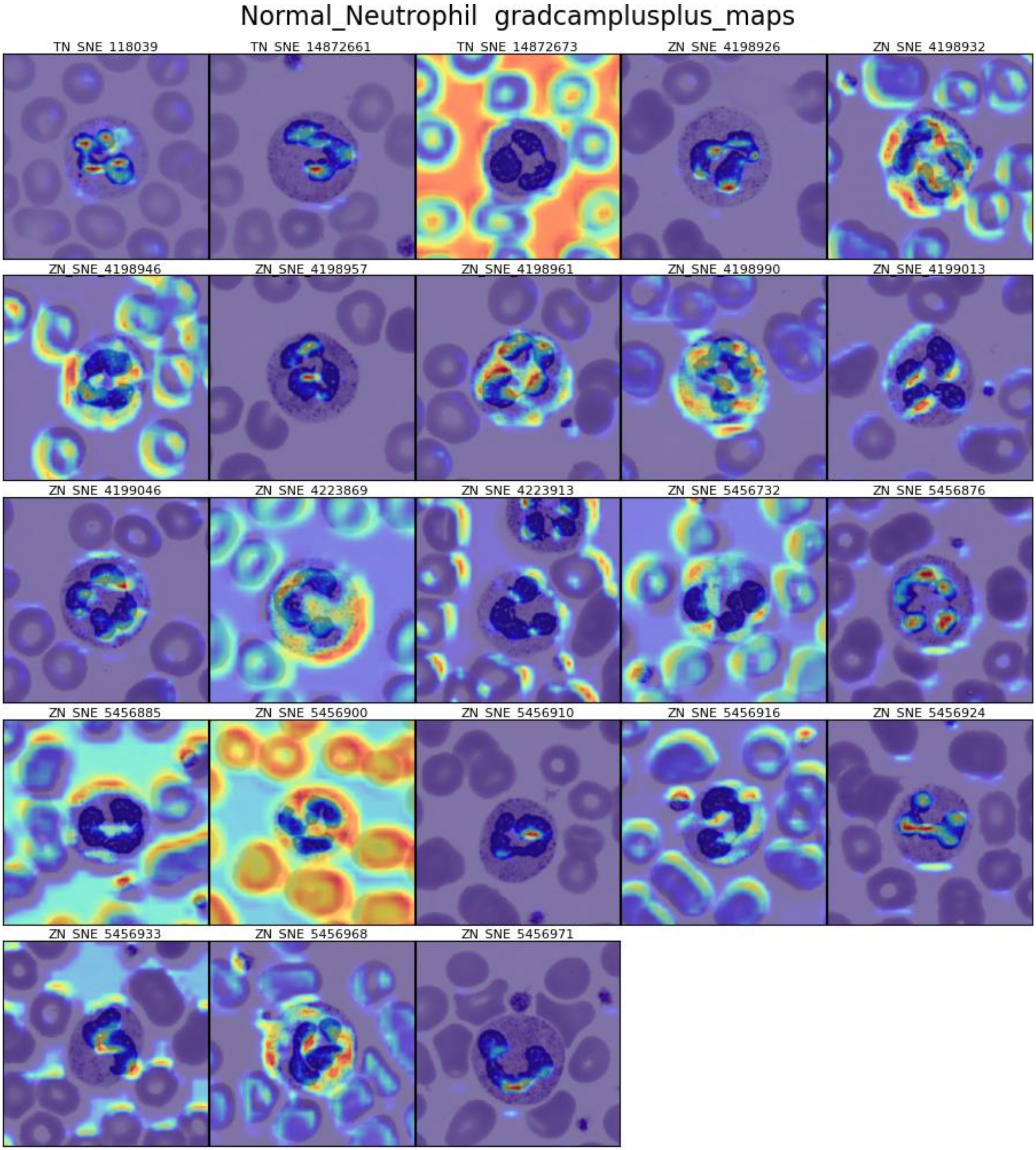

Normal_Neutrophil gradcam_maps
TN SNE 118039
TN SNE 14872661
TN SNE 14872673
ZN SNE 4198926
ZN SNE 4198932
ZN_SNE_4198946
ZN_SNE_4198957
ZN_SNE_4198961
ZN_SNE_4198990
ZN_SNE_4199013
ZN SNE 4199046
ZN SNE 4223869
ZN SNE 4223913
ZN SNE 5456732
ZN SNE 5456876
ZN SNE 5456885
ZN SNE 5456900
ZN SNE 5456910
ZN SNE 5456916
ZN SNE 5456924
ZN SNE 5456933
ZN SNE 5456968
ZN SNE 5456971

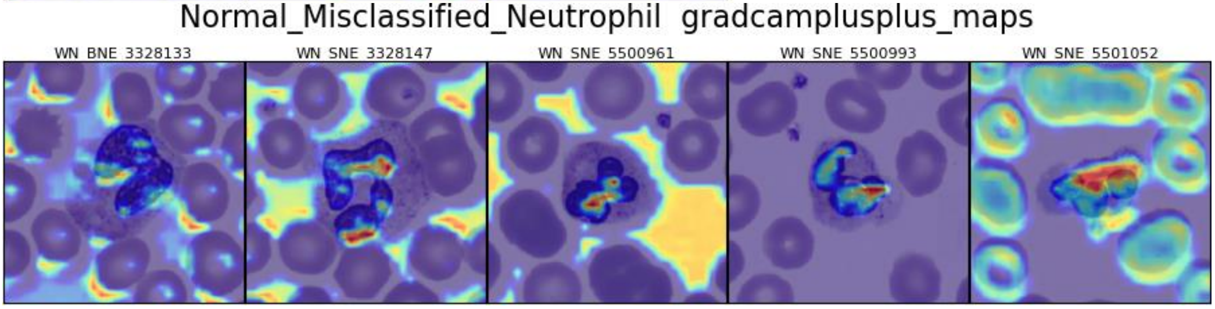

Normal_Misclassified_Neutrophil gradcam_maps
WN BNE 3328133
WN SNE 3328147
WN SNE 5500961
WN SNE 5500993
WN SNE 5501052

C7. Grad-CAM++ Maps



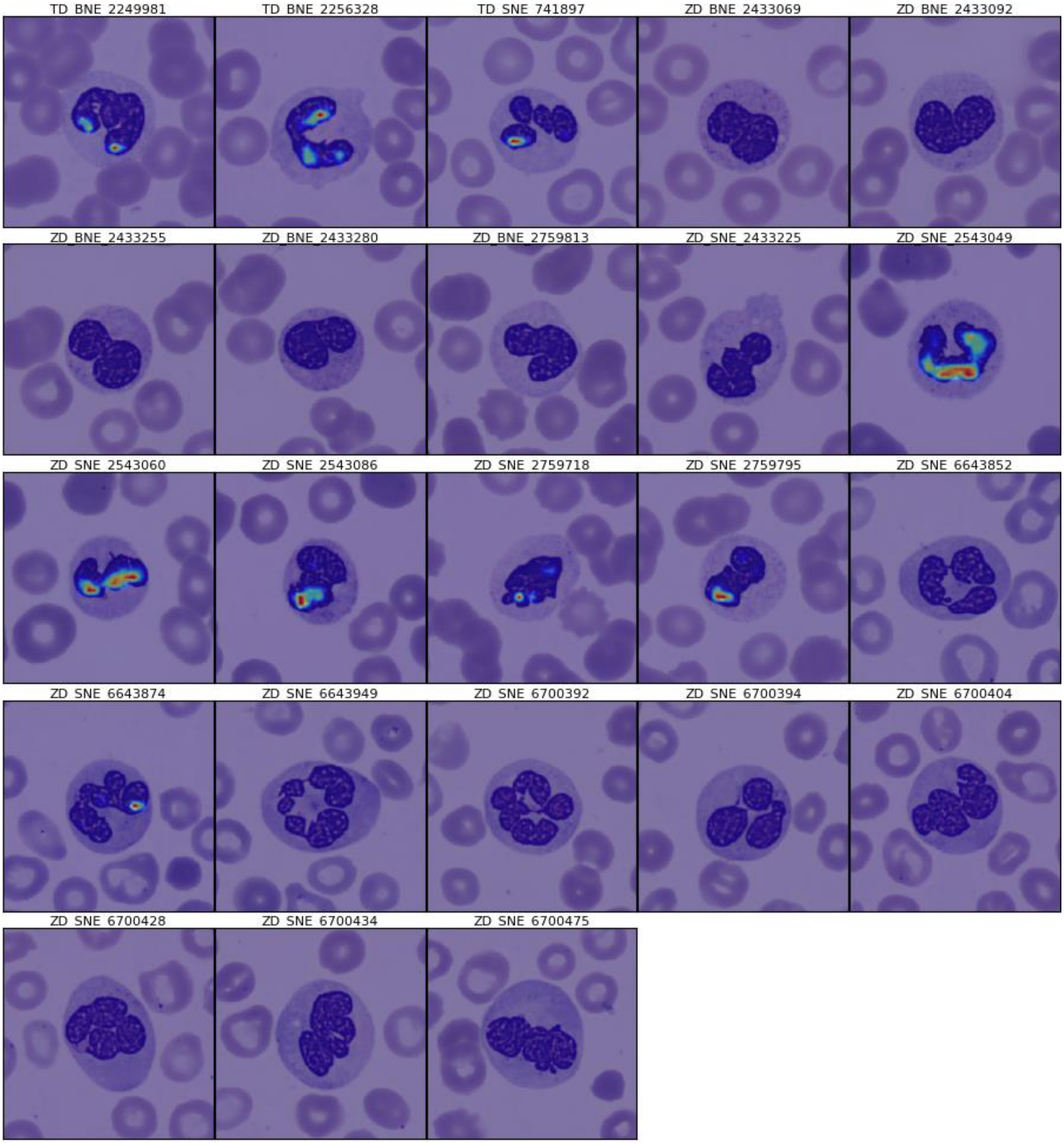



Normal_Neutrophil gradcamplusplus_maps
TN_SNE_118039
TN_SNE_14872661
TN_SNE_14872673
ZN_SNE_4198926
ZN_SNE_4198932
ZN_SNE_4198946
ZN_SNE_4198957
ZN_SNE_4198961
ZN_SNE_4198990
ZN_SNE_4199013
ZN_SNE_4199046
ZN_SNE_4223869
ZN_SNE_4223913
ZN_SNE_5456732
ZN_SNE_5456876
ZN_SNE_5456885
ZN_SNE_5456900
ZN_SNE_5456910
ZN_SNE_5456916
ZN_SNE_5456924
ZN_SNE_5456933
ZN_SNE_5456968
ZN_SNE_5456971

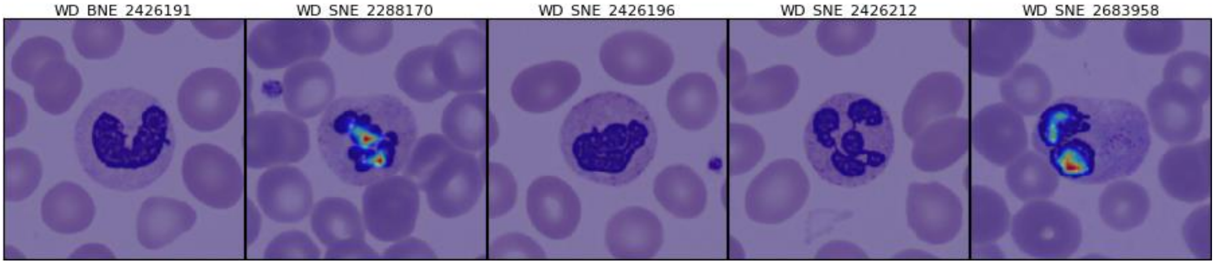

Normal_Misclassified_Neutrophil gradcamplusplus_maps
WN_BNE_3328133
WN_SNE_3328147
WN_SNE_5500961
WN_SNE_5500993
WN_SNE_5501052

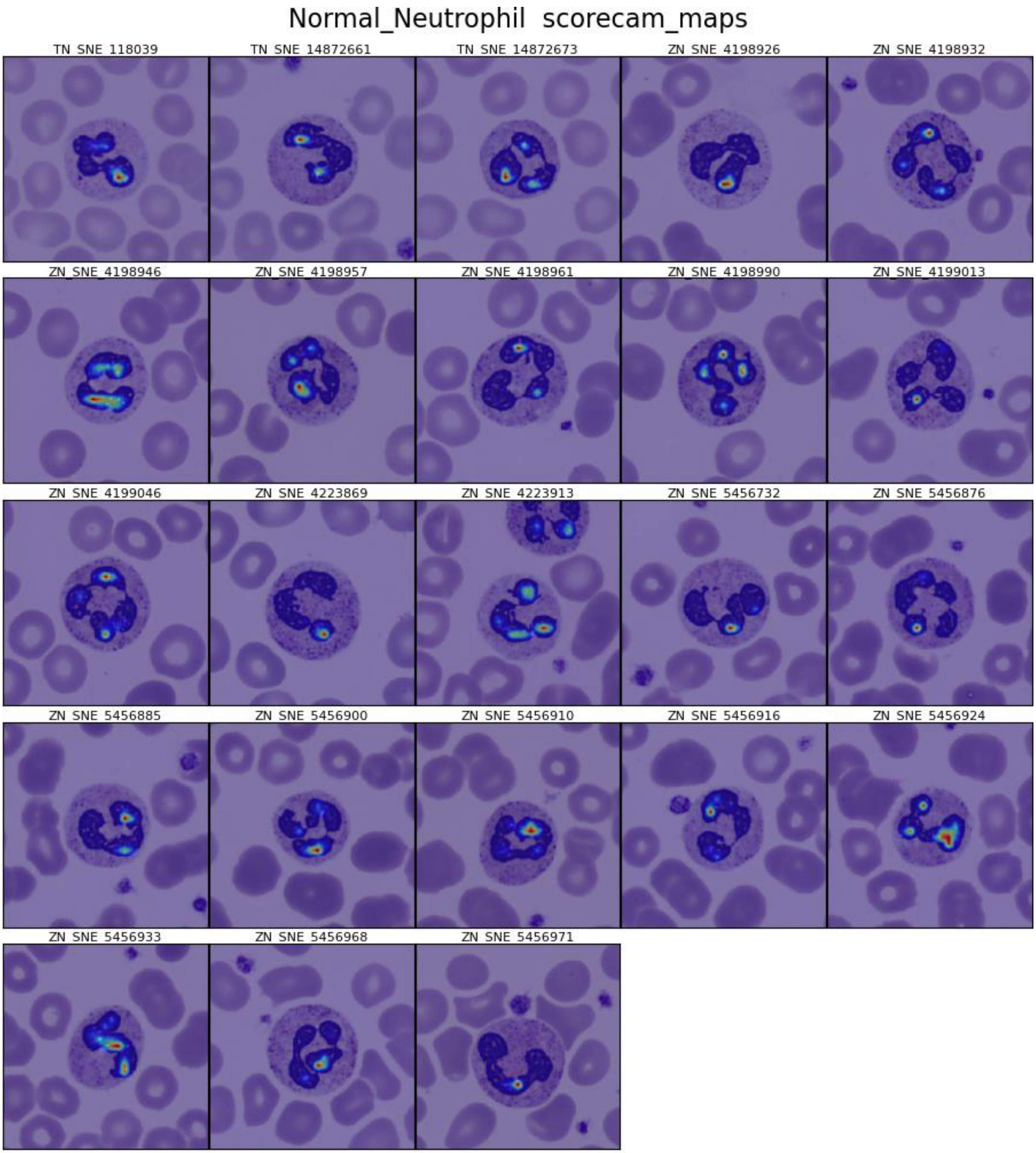
C8. Score-CAM Maps

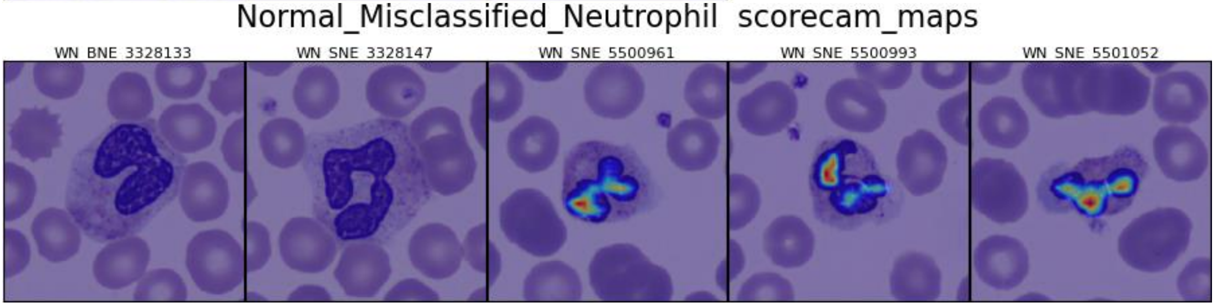
Dysplastic_Neutrophil scorecam_maps



Dysplastic_Misclassified_Neutrophil scorecam_maps




Normal_Neutrophil scorecam_maps
TN_SNE_118039
TN_SNE_14872661
TN_SNE_14872673
ZN_SNE_4198926
ZN_SNE_4198932
ZN_SNE_4198946
ZN_SNE_4198957
ZN_SNE_4198961
ZN_SNE_4198990
ZN_SNE_4199013
ZN_SNE_4199046
ZN_SNE_4223869
ZN_SNE_4223913
ZN_SNE_5456732
ZN_SNE_5456876
ZN_SNE_5456885
ZN_SNE_5456900
ZN_SNE_5456910
ZN_SNE_5456916
ZN_SNE_5456924
ZN_SNE_5456933
ZN_SNE_5456968
ZN_SNE_5456971


Normal_Misclassified_Neutrophil scorecam_maps
WN_BNE_3328133
WN_SNE_3328147
WN_SNE_5500961
WN_SNE_5500993
WN_SNE_5501052

Annex D. Results of the Background Generalization Test

Dysplastic Neutrophils				
Image	Class	Original Score	Masked Score	Difference
TD_BNE_2249981	1.0000	0.0334	0.0001	0.0334
TD_BNE_2256328	1.0000	0.0463	0.0154	0.0309
TD_SNE_741897	1.0000	0.0009	0.0007	0.0002
ZD_BNE_2433069	1.0000	0.3919	0.3793	0.0127
ZD_BNE_2433092	1.0000	0.0013	0.0003	0.0010
ZD_BNE_2433255	1.0000	0.2043	0.0057	0.1986
ZD_BNE_2433280	1.0000	0.0017	0.0198	-0.0181
ZD_BNE_2759813	1.0000	0.0002	0.0009	-0.0007
ZD_SNE_2433225	1.0000	0.1755	0.2909	-0.1154
ZD_SNE_2543049	1.0000	0.0619	0.1329	-0.0710
ZD_SNE_2543060	1.0000	0.0001	0.0070	-0.0069
ZD_SNE_2543086	1.0000	0.0000	0.0005	-0.0005
ZD_SNE_2759718	1.0000	0.0490	0.4975	-0.4485
ZD_SNE_2759795	1.0000	0.0037	0.0384	-0.0347
ZD_SNE_6643852	1.0000	0.0000	0.0000	0.0000
ZD_SNE_6643874	1.0000	0.0000	0.0000	0.0000
ZD_SNE_6643949	1.0000	0.0007	0.0014	-0.0007
ZD_SNE_6700392	1.0000	0.0000	0.0000	0.0000
ZD_SNE_6700394	1.0000	0.0000	0.0001	0.0000
ZD_SNE_6700404	1.0000	0.0000	0.0001	-0.0001
ZD_SNE_6700428	1.0000	0.0000	0.0031	-0.0030
ZD_SNE_6700434	1.0000	0.0001	0.0001	0.0000
ZD_SNE_6700475	1.0000	0.0000	0.0001	-0.0001

Misclassified Dysplastic Neutrophils				
Image	Class	Original Score	Masked Score	Difference
WD_BNE_2426191	3.0000	0.9889	0.7378	0.2511
WD_SNE_2288170	3.0000	1.0000	0.9961	0.0039
WD_SNE_2426196	3.0000	0.9997	0.9995	0.0002
WD_SNE_2426212	3.0000	1.0000	1.0000	0.0000
WD_SNE_2683958	3.0000	1.0000	0.8784	0.1216

Normal Neutrophils				
Image	Class	Original Score	Masked Score	Difference
TN_SNE_118039	2.0000	1.0000	0.9943	0.0057
TN_SNE_14872661	2.0000	0.9999	0.9979	0.0020
TN_SNE_14872673	2.0000	1.0000	0.9975	0.0024
ZN_SNE_4198926	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4198932	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4198946	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4198957	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4198961	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4198990	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4199013	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4199046	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4223869	2.0000	1.0000	1.0000	0.0000
ZN_SNE_4223913	2.0000	0.9916	0.9998	-0.0082
ZN_SNE_5456732	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456876	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456885	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456900	2.0000	0.9998	1.0000	-0.0002
ZN_SNE_5456910	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456916	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456924	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456933	2.0000	0.9998	0.9999	-0.0001
ZN_SNE_5456968	2.0000	1.0000	1.0000	0.0000
ZN_SNE_5456971	2.0000	1.0000	1.0000	0.0000

Misclassified Normal Neutrophils				
Image	Class	Original Score	Masked Score	Difference
WN_BNE_3328133	4.0000	0.0126	0.5017	-0.4891
WN_SNE_3328147	4.0000	0.0541	0.5017	-0.4476
WN_SNE_5500961	4.0000	0.5496	0.8147	-0.2651
WN_SNE_5500993	4.0000	0.0014	0.3207	-0.3193
WN_SNE_5501052	4.0000	0.0001	0.2980	-0.2980