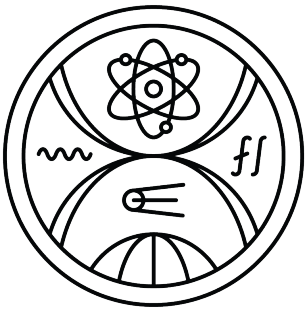
COMENIUS UNIVERSITY IN BRATISLAVA
FACULTY OF MATHEMATICS PHYSICS AND INFORMATICS

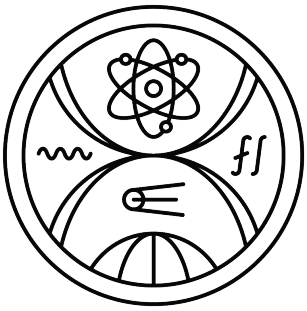


TEXTURAL DESCRIPTORS FOR QUANTIFICATION OF MITOCHONDRIAL STATES

Master thesis

2026 Bc. Matúš Kočalka

COMENIUS UNIVERSITY IN BRATISLAVA
FACULTY OF MATHEMATICS PHYSICS AND INFORMATICS



TEXTURAL DESCRIPTORS FOR QUANTIFICATION OF MITOCHONDRIAL STATES

Master thesis

Study program:	Applied informatics
Branch of study:	Applied informatics
Department:	Department of Applied Informatics
Supervisor:	Dr. Jiří Hladůvka
Consultant:	

Bratislava, 2026 Bc. Matúš Kočalka

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Matúš Kočalka
Študijný program: aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: anglický
Sekundárny jazyk: slovenský

Názov: Textural descriptors for quantification of mitochondrial states
Textúrne deskriptory pre kvantifikáciu stavov mitochondrií

Anotácia: Preskúmajte využitie textúrnych descriptorov odvodených z analýzy obrázkov na kvantifikáciu stavov mitochondrií, ktoré poskytujú náhľad do ich morfologických alebo funkčných charakteristík. Využitím pokročilých výpočtových techník sa snaží premostiť priepasť medzi mikroskopickým zobrazovaním a kvantitatívnou biológiou. Práca sa zameriava na identifikáciu vzorcov a metrík, ktoré korelujú so zdravím alebo dynamikou mitochondrií.

Cieľ:
• Vytvoriť súbor textúrnych descriptorov: Vytvoriť a vyhodnotiť deskriptory prispôsobené na analýzu textúr mitochondrií v mikroskopických obrázkoch.
• Korelovať deskriptory s biologickými stavmi: Stanoviť korelácie medzi vypočítanými textúrnymi vlastnosťami a známymi stavmi mitochondrií, ako je fragmentácia, predĺženie alebo morfológia vyvolaná stresom.
• Navrhnúť pracovný postup pre kvantitatívnu analýzu: Navrhnúť reprodukovateľnú pipeline pre analýzu mitochondriálnych obrázkov, ktorá integruje textúrne deskriptory s biologickou interpretáciou.
• Validácia: Overiť pomocou experimentálnych dátových súborov.

Literatúra:
[1] M. K. Ghalati, A. Nunes, H. Ferreira, P. Serranho, and R. Bernardes, ‘Texture Analysis and Its Applications in Biomedical Imaging: A Survey’, IEEE Rev. Biomed. Eng., vol. 15, pp. 222–246, 2022, doi: 10.1109/RBME.2021.3115703.

[2] I. M. G. M. Hemel, B. P. H. Engelen, N. Luber, and M. Gerards, ‘A hitchhiker’s guide to mitochondrial quantification’, Mitochondrion, vol. 59, pp. 216–224, Jul. 2021, doi: 10.1016/j.mito.2021.06.005.

[3] A. Humeau-Heurtier, ‘Texture Feature Extraction Methods: A Survey’, IEEE Access, vol. 7, pp. 8975–9000, 2019, doi: 10.1109/ACCESS.2018.2890743.

Vedúci: Dr. Jiří Hladůvka, Dr.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: doc. RNDr. Tatiana Jajcayová, PhD.

Spôsob sprístupnenia elektronickej verzie práce:
bez obmedzenia

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Dátum zadania: 09.12.2024

Dátum schválenia: 09.12.2024 prof. RNDr. Roman Ďurikovič, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

I hereby declare that I have written this thesis by myself, only with help of referenced literature, under the careful supervision of my thesis advisor.

.....

Bratislava, 2026 Bc. Matúš Kočalka

Acknowledgement

Abstract

Mitochondria are among the most crucial organelles in the cell. Their morphology is often interpreted as a direct indicator of cellular demands, which are shaped by the surrounding environment. In recent years, changes in mitochondrial shape have been linked to a number of diseases, including cancer and diabetes.

Because of these relationships, the study of mitochondrial structure and dynamics has become an increasingly promising field in medicine and biology. Consequently, many tools and procedures have been developed to enable fully and semi-automated, quantitative analysis of mitochondrial morphology from microscopy images.

In our work, we propose a new approach for analyzing and quantifying mitochondrial states, with a strong emphasis on interpretability. We leverage the rich domain of established texture features, aiming to sufficiently characterize mitochondrial states using a limited, carefully selected subset of these features. This subset is automatically chosen from a broad pool of texture descriptors, based on a small set of annotated examples.

Keywords: mitochondria, textural features, quantization

Abstrakt

Mitochondrie patria medzi najdôležitejšie organely v bunke. Ich morfológia sa často interpretuje ako priamy ukazovateľ bunkových požiadaviek, ktoré sú formované okolitým prostredím. V posledných rokoch boli zmeny v tvare mitochondrií spojené s viacerými ochoreniami, vrátane rakoviny a cukrovky.

Vďaka týmto súvislostiam sa štúdium štruktúry a dynamiky mitochondrií stalo čoraz perspektívnejšou oblasťou v medicíne a biológii. V dôsledku toho bolo vyvinutých mnoho nástrojov a postupov, ktoré umožňujú automatizovť, kvantitatívnu analýzu mitochondriálnej morfológie z mikroskopických snímok.

V našej práci navrhujeme nový prístup k analýze a kvantifikácii stavu mitochondrií s dôrazom na interpretovateľnosť. Využívame bohatú oblasť známych textúrnych príznakov s cieľom dostatočne charakterizovať stav mitochondrií pomocou obmedzenej, starostlivo vybranej podmnožiny týchto príznakov. Táto podmnožina je automaticky vybraná zo širokej škály textúrnych deskriptorov na základe malej množiny anotovaných príkladov.

Kľúčové slová: mitochondria, textúrne príznaky, kvantizácia

Contents

List of Figures

List of Tables

Terminology

Terms

Motivation

Chapter 1

Introduction

Cells are the fundamental building blocks of all known living organisms. Living organisms are commonly classified into two groups, prokaryotes and eukaryotes. Prokaryotic organisms are composed of a single cell, which lacks a membrane-bound nucleus, mitochondria, or other complex organelles. On the other hand, Prokaryotes organisms can also be unicellular, but also or multicellular. Their cells are more complex with a true nucleus and many specialized membrane-bounded systems called organelles. These organelles perform various specialized, vital functions.

One such organelle is the mitochondrion. Mitochondria are one of the most crucial organelles in a cell and are commonly referred to as the "powerhouses of the cell." It's because one of the main functions of mitochondria is generating the majority of a cell's chemical energy, ATP production. Other functions that mitochondria are responsible for include calcium signaling, regulation of oxidative metabolism, and control of programmed cell death.

Mitochondria have some unique features. Mitochondria have distinct inner and outer membranes. They also have their own mtDNA. And, for us most interesting feature, their dynamic nature. Mitochondria are constantly changing their shape to better adapt their function, in order to better satisfy cells' needs.

Mitochondrial morphology can then be interpreted as a direct indicator of cellular demands. These demands are formed by the outer environment. In addition, in recent years, many diseases, like cancer and diabetes, for example, have been associated with mitochondrial shape changes.

It is precisely because of these relationships that the study of mitochondria is a much more promising field in medicine and biology. For this reason, many tools and procedures have been developed to facilitate automated quantitative analysis of mitochondrial morphology from microscopy images. We will cover those approaches in section Related work **??**.

Mitochondria can possess extremely diverse morphological states ranging from ex-

ceedingly complex networks to sparse, fractured blobs. And thus its extremely hard to define a finite set of specific features that can describe all possible morphological states sufficiently.

In our work, we propose our own approach on how to analyze and quantify the state of mitochondria, with great focus on iterpretability (necessary biologically meaningful from the start) of such anlaysis. We took advantage of the rich field of known texture features, in the hope that we could sufficiently quantify the mitochondria state with a limited set of chosen features. Where this set of features is automatically selected from a broad set of texture features, based on a small dataset of examples.

More about texture features that we adapt in our work can be found in section 1.1.

1.1 Texture features

For human, cognition of visual textures feel completely natural. Our brains can easily distinguish regions that differ only in their surface structure, patterns, or granularity. In contrast, formally defining what texture is or how it should be measured has repeatedly proven to be a challenging problem in computer vision.

We can easily define texture as a pattern of image intensities together with their spatial distribution. However, this definition is extremely broad and non-specific. The possible combinations of intensity values and spatial relationships are so diverse that such texture definition offers no practical solution to how to extract a useful texture feature. As a result, such a definition does not provide a clear guideline for extracting texture features from an image.

Because of this inherent ambiguity, a wide range of mathematical and statistical approaches have been proposed to quantify textural properties. Researchers have derived numerous feature descriptors ranging from simple measures of intensity variation to sophisticated models of spatial correlation, in an attempt to capture what texture "means" in a computational sense.

Over time, many major families of texture descriptors have been developed. Classical methods include statistical features such as grey-level co-occurrence matrices (GLCM), model-based approaches like Markov random fields, and transform-based representations such as wavelets or Gabor filters. More recently, deep learning techniques have introduced high-level, learned texture embeddings that often outperform hand-crafted features. Yet despite these advances, there is still no universally accepted definition of texture, and the choice of descriptor often depends on the specific task, domain, and computational constraints.

Most commonly used texture features can be grouped into the following six main families:

- Grey level co-occurrence based features (GLCM)
- Grey level run length based features (GLRLM)
- Grey level size zone based features (GLSZM)
- Grey level distance zone based features (GLDZM)
- Neighbourhood grey tone difference based features (NGTDM)
- Neighbouring grey level dependence based features (NGLDM)

These feature families are often referred to as higher-order texture features, because they describe spatial relationships between two or more voxel intensities in the image. Despite each family expressing different properties of texture, they share several important characteristics.

For every texture feature family, the computation begins by constructing a predefined texture matrix. The method used to build this matrix differs between families, as each captures a different type of spatial relationship (voxel pairs, runs, zones, neighbourhood differences, or local dependencies). However, for a large subset of features, this matrix definition is the only thing that changes.

Once the matrix is constructed, many texture features are computed by applying the same statistical formulas, such as emphasis measures, non-uniformity measures, energy, entropy, or grey level based statistics. In these cases, the formulas are identical across families; only the underlying matrix differs.

Some texture features are specific to only one or a few families, typically because they rely on statistics or structural properties that exist only in the corresponding matrix. As a result, not all features have direct equivalents across the different texture families, but many of them do share common mathematical patterns.

In the following subsection 1.1.1, we describe in detail, for each family, how the texture matrix is computed from image intensities. For every feature family, this step defines the spatial relationships that the matrix captures, whether it is pairs of voxel intensities (as in GLCM), consecutive runs of identical values (GLRLM), connected zones (GLSZM), voxel dependences (GLDM), or neighborhood intensity differences (NGTDM). Understanding the construction of these matrices is essential, as they define what property of texture derived measures quantify.

After defining the texture matrices, we turn to the computation of the features themselves. In this subsection 1.1.2, we categorize features into shared formula classes, which have identical mathematical definitions across multiple matrices, and unique features, which exist only in specific families due to the structural properties of the corresponding family matrices.

Together, these two sections provide a complete overview of classical texture analysis. First, the predefined matrices capture the spatial context of voxel intensities, and finally, statistical measures from these matrices are used as quantitative features for downstream analysis.

1.1.1 Texture natrix construction

Texture analysis in radiomics relies on several families of texture matrices, each designed to capture a different aspect of spatial patterns in an image. Although these matrices quantify texture in different ways, pairwise grey-level relationships, run lengths, zones, or neighborhood differences, they all begin with the same fundamental step, such as defining how voxels in the image relate to each other. This requires specifying which voxels are considered neighbors and their mutual distance. In the IBSI standard, this spatial relationship is consistently defined using the Chebyshev distance, δ.

The Chebyshev distance is a metric used to measure the distance between two points in a multidimensional space. It is defined as the maximum absolute difference among their corresponding coordinates. This metric is also known as the maximum metric, or the L_∞ metric.

So if we have 2 points in N-dimensional space: $A = (a_i); 0 \geq i \geq n$ and $B = (b_i); 0 \geq i \geq n$ we can calculate Chebyshev distance as:

$$d(A, B) = max_i(|a_i - b_i|).$$

We can see that in 2D space, for a given voxel, the Chebyshev distance $\delta = 1$ corresponds to an 8-connected neighborhood, which can also be represented as a 3×3 kernel.

For different values of Chebyshev distance, we can calculate multiple texture matrices, achieving a variety of informative values.

In the following subsections, we will discuss how individual texture matrices are defined and show simple examples of how they are calculated.

Grey level co-occurrence matrix

The Grey-Level Co-Occurrence Matrix (GLCM) describes image texture by calcul ating the distribution for each pair of pixel intensity values in the image in each direction. Unlike first-order statistics, which only consider the distribution of grey levels independently (such as histograms), the GLCM captures second-order statistics, how intensities relate to one another across space. This makes it particularly useful when the visual appearance of a region is defined not by absolute brightness values but by repetitive or structured patterns, such as smoothness, granularity, or regularity.

The Image Biomarker Standardisation Initiative (IBSI) [4] recommends computing the GLCM in each unique direction of a 2D 8-connected neighborhood. In two dimensions, this results in four non-redundant directions: left(0°), left-up(45°), up(90°), and right-up(135°).

We can exclude the opposite direction, for example, the direction left corresponding to 0°, and its opposite direction, right, corresponding to 180° degrees, would both result in a transposed version of each other. So, opposite directions do not contain additional texture information. In GLCM compuation example 1.1 we can see, that (c) matrix, corresponding to 180° direction is transposed version of (b) matrix, corresponding to opposite, 0° direction. For this reason, using only the four primary directions is sufficient to fully characterize a 2D 8-connected neighborhood, while avoiding redundant computation and enabling approximate rotational invariance.

For a given Chebyshev distance $\delta = 1$, we can encode those directions by the offset vectors: $D = \{(1,0),(1,1),(0,1),(-1,1)\}$. For different distances, we just need to multiply those vectors by δ, and the subsequent GLCM computation remains unchanged.

The computation of a GLCM for a given direction m and distance δ proceeds as follows. Initialize a square matrix M_{+m} with shape $n \times n$ to zeroes, where n is the number of unique intensities in image I and $+m \in D$. Then for each observed pair of intensities i, j in direction m in the image I. Increase matrix entry $M_{+m}(i,j)$ by one.

As mentioned before, the opposite direction $-m$ is symmetrical to this calculated matrix $M_{+m}(i,j)$, Therefore, it is unnecessary to compute it separately. Instead, the symmetric GLCM can be obtained by

$$M_m = M_{+m} + M_{-m} = M_{+m} + M_{+m}^T.$$

The resulting symmetric matrix M_m incorporates co-occurrence information from both directions $+m$ and $-m$, ensuring that the GLCM captures all spatial relationships along that axis without redundant computation. Example of this precess can be seen in table 1.1.

With M_m matrix caluclated for each of four primary directions D. we can convert them into probability distribution by dividing each element by the total number of counted voxel/pixel pairs:

$$P_m(i,j) = \frac{M_m(i,j)}{\sum_{k=1}^{n}\sum_{l=1}^{n} M_m(k,l)}$$

This way we get actual GLCM matrices used to caluclate features.

3	4	4	4
1	4	2	1
3	2	1	1
3	2	3	2

(a) Grey levels, image I

		j			
		1	2	3	4
i	1	1	0	0	1
	2	2	0	1	0
	3	0	3	0	1
	4	0	1	0	2

(b) M_{+m}

		j			
		1	2	3	4
i	1	1	2	0	0
	2	0	0	3	1
	3	0	1	0	0
	4	1	0	1	2

(b) M_{-m}

		j			
		1	2	3	4
i	1	2	2	0	1
	2	2	0	4	1
	3	0	4	0	1
	4	1	1	1	4

(b) M_m

Table 1.1: Example of computing GLMC matrix M_m in direction $+m = (1, 0)$, left($0°$) and Chebyshev distance, $\delta = 1$. (a) Grey level input image, with four unique intensity values. (b) corresponding GLCM matrix in direction $+m = (1, 0)$. (c) Example of corresponding GLCM in opposite direction, $-m = (-1, 0)$, right left($180°$). (d) Final symmetrical GLCM matrix.

Grey level run length matrix

The Grey-Level Run Length Matrix (GLRLM) is a texture descriptor that represents how pixels of identical intensity (gray level) form contiguous, linear structures, called runs, within an image. A gray-level run is defined as an unbroken sequence of pixels that share the same intensity in a given direction.

Similarly to the Grey-Level Co-Occurrence Matrix (GLCM), see Section 1.1.1, the GLRLM is computed separately for each direction D. But, unlike the GLCM, there is no need to compute the matrix for the opposite direction. Because a run is a contiguous sequence of same-intensity pixels, its length does not depend on whether it is traversed from start to end or end to start. So the run length is the same in both directions, and thus, opposite directions provide no additional information.

The first step in constructing the GLRLM for a given direction mmm is to initialize a matrix M_m of size $n \times r$ with zeros. Where n is the number of discrete

gray levels present in the processed image, and r is the maximum possible run length in direction m. Each matrix element $M_m(i, j)$ then counts the number of runs of length j consisting of pixels with gray level i. An example of such a GLRLM is shown in Table 1.2.

3	4	4	4
1	4	2	1
3	2	1	1
3	2	3	2

(a) Grey levels, image I

		j			
		1	2	3	4
i	1	4	1	0	0
	2	4	0	0	0
	3	4	0	0	0
	4	4	2	1	0

(b) GLRM in direction m

Table 1.2: Example of GLRM. (a) input, grey level image. (b) Corresponding GLRM in direction (1,0), left (0°).

Grey level size zone matrix

The GLSZM quantifies texture in a grayscale image by counting connected zones of the same gray level intensity. A connected zone or area is determined by connectiveness. Connectivity describes whether a pixel is connected to another. As mentioned in IBS [4]: In the 2 dimensional approach, 8-connectedness is used. This means that for a given pixel with some intensity i, if in its 8-neighborhood is another pixel with the same intensity, then both of them belong to the same zone.

In contrast to the previously discussed matrices—GLCM and GLRLM, which are direction-dependent—the GLSZM is direction-independent. Therefore, only a single matrix is computed. The first step is to initialize a zero matrix M of size $n \times s$, where n is the number of unique gray levels in the image and s is the maximum zone size present. Each matrix entry $M(i, j)$ records the number of zones with gray level i and size j.

An example illustrating how this matrix is constructed is shown in Table 1.3.

3	4	4	4
1	4	2	1
3	2	1	1
3	2	3	2

(a) Grey levels, image I

		j			
		1	2	3	4
i	1	1	0	1	0
	2	1	0	1	0
	3	2	1	0	0
	4	0	0	0	1

(b) $GLSZM$

Table 1.3: Example of GLSZM. (a) input, grey level image. (b) Corresponding GLSZM.

Grey level distance zone matrix

The Grey Level Distance Zone Matrix (GLDZM) is conceptually similar to the Grey Level Size Zone Matrix (GLSZM). However, instead of recording the size of connected zones, the GLDZM records how far each zone is from the boundary of the region of interest (ROI). The first steps of the computation are the same as for the GLSZM.

The image or the ROI is first partitioned into zones of connected pixels that share the same intensity. In 2D, this is typically defined using 8-connectivity; see section 1.1.1.

Next, we need to define the distance of each zone. IBSI [4] defines distance maps using 4-connectivity in 2D. For each zone, its distance is defined as the minimum number of steps required to reach the boundary of the ROI, where each step moves to a 4-connected neighbor. Importantly, this minimum is taken over all pixels belonging to the zone.

This distance can be computed using a distance map, where each pixel stores its distance to the ROI boundary. Let $DM(i, j)$ denote the distance of pixel (i, j). For a zone Z, consisting of pixels (k, l) that share the same grey level and are connected, the zone distance is:

$$zone\_distance(z) = min_{(k,l)\in Z}(DM(k, l)).$$

The distance map itself can be obtained iteratively. Starting with the ROI mask, initialize a distance image with zeros. All pixels inside the ROI are assigned a value of 1. Then, erode the ROI mask using 4-connected structuring elements; after each erosion step, increment the distance value for pixels still inside the mask. Repeating the erosion and increment steps until the ROI has completely eroded results in a complete distance map.

Once the zones and their distances are known, constructing the GLDZM is straightforward. We initialize a matrix M of size $n \times d$, where n is the number of unique discretized grey levels and d is the maximum observed zone distance. Each entry $M(i, j)$ counts the number of zones with grey level i and distance j. An example GLDZM is provided in Table 1.4.

3	4	4	4
1	4	3	1
3	2	1	1
3	2	3	2

(a) Grey levels, image I

1	1	1	1
1	2	2	1
1	2	2	1
1	1	1	1

(b) Distance map

		j	
		1	2
i	1	2	0
	2	2	0
	3	3	1
	4	1	0

(c) GLDZM

Table 1.4: Example of GLDZM. (a) input, grey level image. (b) distance map, note that ROI is in this case area of whole image. (c) corresponding GLDZM.

Neighbourhood grey tone difference matrix

The neighbourhood grey tone difference matrix (NGTDM) quantifies image texture by capturing the difference in intensity between each pixel and its neighbours within a Chebyshev distance δ. NGTDM is somewhat unique in the texture matrix families, as it does not compute a metric for gradually increasing parameters (run length or zone size). Instead, the neighbourhood grey tone difference matrix computes 2 values for each intensity present in the grey level image.

The first value is probability.

$$p_i = \frac{n_i}{\sum_i n_i}$$

Where n_i is the number of pixels with the same intensity i in ROI and $\sum_i n_i$ is the total pixel count in ROI.

The second value is mentioned difference in intensity between each pixel and its neighbours.

$$s_i = \sum_k |i - \bar{X}_k|[X_d(k) = i \wedge W_k > 0]$$

Where $k = (k_x, k_y)$ is one pixel from the image. Notation [...] means a condition that, if satisfied, returns 1 else 0. $X_d(k)$ is function that returns intensity for input pixel k. $\bar{X}_k$ is mean intensity of neighbourhoof of pixel k. and can by computed as fllows:

$$\bar{X_k} = \frac{1}{W_k} \sum_{m_x=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} X_d(k + (m_x, m_y))[(m_x, m_y) \neq (0,0) \wedge k + (m_x, m_y) \in ROI]$$

And finally W_k is total area of neighbourhood.

$$W_k = \sum_{m_x=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} [(m_x, m_y) \neq (0,0) \wedge k + (m_x, m_y) \in ROI]$$

Note that the neighbourhood doesn't need to be complete. For example, in 2D for a given Chebyshev distance δ, W_k is upperbounded by $W_k = (2\delta + 1)^2 - 1$, but can be lower, or even 0, as not each pixel must be inside the ROI or Image whatsoever.

So in summary, NGTDM calculates, for each intensity, the total average difference of each pixel with given intensity and a non-zero neighbourhood.

An example of calculated NGTDM can be seen in the Table 1.5.

Neighbouring grey level dependence matrix

The Neighbouring Grey Level Dependence Matrix (NGLDM) is a statistical texture matrix that quantifies how many neighbouring pixels in a grayscale image are dependent on a given centre pixel based on their grey-level similarity and spatial proximity.

3	4	4	4
1	4	3	1
3	2	1	1
3	2	3	2

(a) Grey levels, image I

3	5	5	3
5	8	8	5
5	8	8	5
3	5	5	3

(b) W_k map

$\frac{9}{3} = 3$	$\frac{15}{5} = 3$	$\frac{16}{5} = 3.2$	$\frac{8}{3} = 2.\bar{6}$
$\frac{16}{5} = 3.2$	$\frac{21}{8} = 2.625$	$\frac{21}{8} = 2.625$	$\frac{13}{5} = 2.6$
$\frac{12}{5} = 2.4$	$\frac{20}{8} = 2.5$	$\frac{18}{8} = 2.25$	$\frac{10}{5} = 2$
$\frac{7}{3} = 2.\bar{3}$	$\frac{12}{5} = 2.4$	$\frac{8}{5} = 1.6$	$\frac{5}{3} = 1.\bar{6}$

(b) $\bar{X_k}$ map

		p_i	s_i
i	1	0.25	6.05
	2	0.1875	$0.4\bar{3}$
	3	0.3125	$0.041\bar{6}$
	4	0.25	0 $4.508\bar{3}$

(c) GLDZM

Table 1.5: Example of NGTDM . (a) input, grey level image. (b) W_k map, note that ROI is in this case area of whole image. (c) $\bar{X_k}$ map, caluclated from (a) and (b). (d) Final GLDZM. For example $p_1 = 4/16 = 0.25$ and $s_1 = |1 - 3.2| + |1 - 2.25| + |1 - 2.6| + |1 - 2| = 6.05$

It was introduced as an alternative to the Grey Level Co-occurrence Matrix (GLCM) to capture texture coarseness and local dependence patterns rather than only pairwise co-occurrences.

The neighbourhood of a given pixel is defined using the Chebyshev distance δ. All pixels whose Chebyshev distance to the centre pixel is less than or equal to δ belong to its neighbourhood. Two pixels in the same neighbourhood are considered dependent if their intensity difference is smaller than the coarseness parameter α. The dependence count j_k of a pixel $k = (k_x, k_y)$ is defined as

$$j_k = \sum_{m_x=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} [|X_d(k) - X_d(k + (m_x, m_y))| \leq \alpha \wedge k + (m_x, m_y) \in ROI]$$

Where notation [...] denotes an indicator function that returns 1 if the condition is satisfied and 0 otherwise. $X_d()$ is a function that, for a given pixel k, returns its intensity. α is a toonable, non-negative integer coarseness parameter (usually $\alpha = 0$ is used).

Note that the minimum dependence value j_k is always at least 1, because the centre pixel is included in its own neighbourhood. This is done intentionally, as several

NGLDM-based texture features are undefined for dependence values less than 1.

The Neighbouring Grey Level Dependence Matrix is then constructed as follows. Initialize a matrix M of size $n \times dp$ with zeros, where n is the number of quantised grey levels in the image and dp is the maximum dependence count observed. Each matrix element $M(i, j)$ stores the number of neighbourhoods whose centre pixel has intensity i and dependence j.

Example of NGLDM can be examined in Table 1.6.

1	2	2	3
1	2	3	3
4	2	4	1
4	1	2	3

(a) Grey levels, image I

2	3	3	3
2	4	3	3
2	3	1	1
2	1	2	1

(b) dependences map I

		dependence			
		1	2	3	4
i	1	0	0	0	0
	2	0	0	1	1
	3	0	0	1	0
	4	1	0	0	0

(c) GLDZM

Table 1.6: Example of NGLDM. (a) input, grey level image. (b) calculated s_k dependeces for each pixel k, with coarseness parameter $\alpha = 0$ and chebyshev distance $\delta = 1$. (c) resulting NGLDM.

1.1.2 Computation of texture features

In previus subsection 1.1.1 we discusssed how to compute individual matrices, that captures unique properties of image texture. But these matrices are just first step in actual feature extraction, as we need to express individual features as single real number.

Great number of features uses marginal matrix values. For given matrix m, those are:

- row marginal sum $m_{.j} = \sum_{i=1}^{N_i} m_{ij}$, where N_i is size of first matrix dimension,
- column marginal sum $m_{i.} = \sum_{j=1}^{N_j} m_{ij}$, where N_j is size of second matrix dimension.

Other used important variables are:

- s sum of all elements of matrix $s = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} m_{ij}$,
- v total number of pixels in ROI,
- $\mu = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} i \frac{m_{ij}}{s}$.

Energy

$$f(m) = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} m_{ij}^2$$

Low range emphasis

$$f(m) = \frac{1}{s} \sum_{j=1}^{N_j} \frac{m_{.j}}{j^2}$$

High range emphasis

$$f(m) = \frac{1}{s} \sum_{j=1}^{N_j} j^2 m_{.j}$$

Low grey level range emphasis

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} \frac{m_{i.}}{i^2}$$

High grey level range emphasis

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} i^2 m_{i.}$$

Low range low grey level emphasis

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \frac{m_{ij}}{i^2 j^2}$$

Low range high grey level emphasis

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \frac{i^2 m_{ij}}{j^2}$$

High range low grey level emphasis

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} \frac{j^2 m_{ij}}{i^2}$$

High range high grey level emphasis

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} i^2 j^2 m_{ij}$$

Grey level non-uniformity

$$f(m) = \frac{1}{s} \sum_{i=1}^{N_i} m_{i.}^2$$

Normalised grey level non-uniformity

$$f(m) = \frac{1}{s^2} \sum_{i=1}^{N_i} m_{i.}^2$$

Range non-uniformity

$$f(m) = \frac{1}{s} \sum_{j=1}^{N_j} m_{.j}^2$$

Normalised range non-uniformity

$$f(m) = \frac{1}{s^2} \sum_{j=1}^{N_j} m_{.j}^2$$

Range percentage

$$f(m) = \frac{s}{v}$$

Grey level variance

$$f(m) = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} (i - \mu)^2 \frac{m_{ij}}{s}$$

Range variance

$$f(m) = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} (j - \mu)^2 \frac{m_{ij}}{s}$$

Range entropy

$$f(m) = -\sum_{i=1}^{N_i} \frac{m_{ij}}{s} \log_2 \frac{m_{ij}}{s}$$

Range energy

$$f(m) = \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} (\frac{m_{ij}}{s})^2$$

Feature Class	GLCM	GLRLM	GLSZM	GLDZM	NGTDM	NGLDM
Energy	✓				✓	
Low range emphasis		✓	✓	✓		✓
High range emphasis		✓	✓	✓		✓
Low grey level range emphasis		✓	✓	✓		✓
High grey level range emphasis		✓	✓	✓		✓
Low range low grey level emphasis		✓	✓	✓		✓
Low range high grey level emphasis		✓	✓	✓		✓
High range low grey level emphasis		✓	✓	✓		✓
High range high grey level emphasis		✓	✓	✓		✓
Grey level non-uniformity		✓	✓	✓		✓
Normalised grey level non-uniformity		✓	✓	✓		✓
Range non-uniformity		✓	✓	✓		✓
Normalised range non-uniformity		✓	✓	✓		✓
Range percentage		✓	✓	✓		✓
Grey level variance		✓	✓	✓		✓
Range variance		✓	✓	✓		✓
Range entropy		✓	✓	✓		✓
Range energy						✓

Table 1.7: Shared texture feature classes across multiple IBSI texture families. The formulas are identical; only the underlying matrix changes.

1.1.3 Discretization

Grey-level discretization is the process of converting continuous or high-resolution intensity values of an image into a smaller set of discrete bins. It is an essential preprocessing step before extracting texture features.

Microscopy images, like many scientific imaging formats, often use a high value range. For example, a .czi file (Carl Zeiss Image) stored in 16-bit format has a theoretical intensity range of 0 to 65,535, meaning potentially up to 65,536 unique grey values. Such a range is far too large to produce meaningful or stable texture features.

One reason is that texture matrices scale with the number of grey levels. If an image truly used all 65,536 values, the corresponding GLCM would have $65,536^2$ elements, which is computationally infeasible. Even more importantly, such matrices become extremely sparse: most grey-level pairs would never occur, leaving the matrix full of zeros and producing unstable or non-informative statistics. In practice, even if a real texture pattern exists, slight pixel-to-pixel intensity fluctuations (noise, illumina-

tion differences, sensor variability) will spread values across many unique grey levels, preventing meaningful co-occurrence counts.

This issue persists even with 8-bit images. A 0–255 intensity range creates a $256^2 = 65,536$ cell GLCM, which is large enough to produce noise-dominated, sparsely populated matrices and highly unstable feature values. Even small amounts of noise can drastically affect the computed texture features when too many grey levels are used.

Grey-level discretization addresses these issues by reducing the number of unique intensities and grouping similar pixel values together. This produces denser matrices, improves computational efficiency, and yields more robust and reproducible texture features.

A commonly used approach is fixed bin number (FBN) discretization. Typical choices range from $n = 16$ or 32 up to 128 bins. In this method, the intensity range of the image is divided into n equally sized bins:

$$bin\_size = \frac{I_{max} - I_{min}}{n}$$

Each pixel is then assigned a new intensity value corresponding to the bin into which its original intensity falls.

Chapter 2

Data

Chapter 3

Implementation

3.1 Segmentation

3.2 Preprocessing

3.3 Feature extarction

3.4 Feature selection

To extract texture features, in depth described in section 1.1 we used Mirp [3] librarry implementing IBS [4].

3.4.1 Feature filtering

3.4.2 search selection

Chapter 4

Research

4.1 Results

In our work we worked with the datasetconsisting of six clasees. Three different enviroment and iamges from two different days for each. Intensity differences in datsaset can be examined in image Figure 4.1. As we can see, only by intenisty, day 1 is simmilar across all enviroments, which corresponds with assumtion, that in first day cells don't have wnough time to respond to enviroment change. But in third day we can examine drastical changes in enviroments SD and YPD, but Acestate stays unchanged.

Resulting classification with seelected features can be seen in Figure 4.2. we tried to classify cells into all 6 classes, witch is in its essence unreasonable task, as each class for day 1 and day 3 Acettate are practically identical. Newertheless in Figure can be clerarly seen clustering of brown cluster, day 3 YPD, red cluster, day 3 SD and rest of the colors clustered in one big cluster in center. witch is actually good sign and actual accuracy is much higher than 0.68.

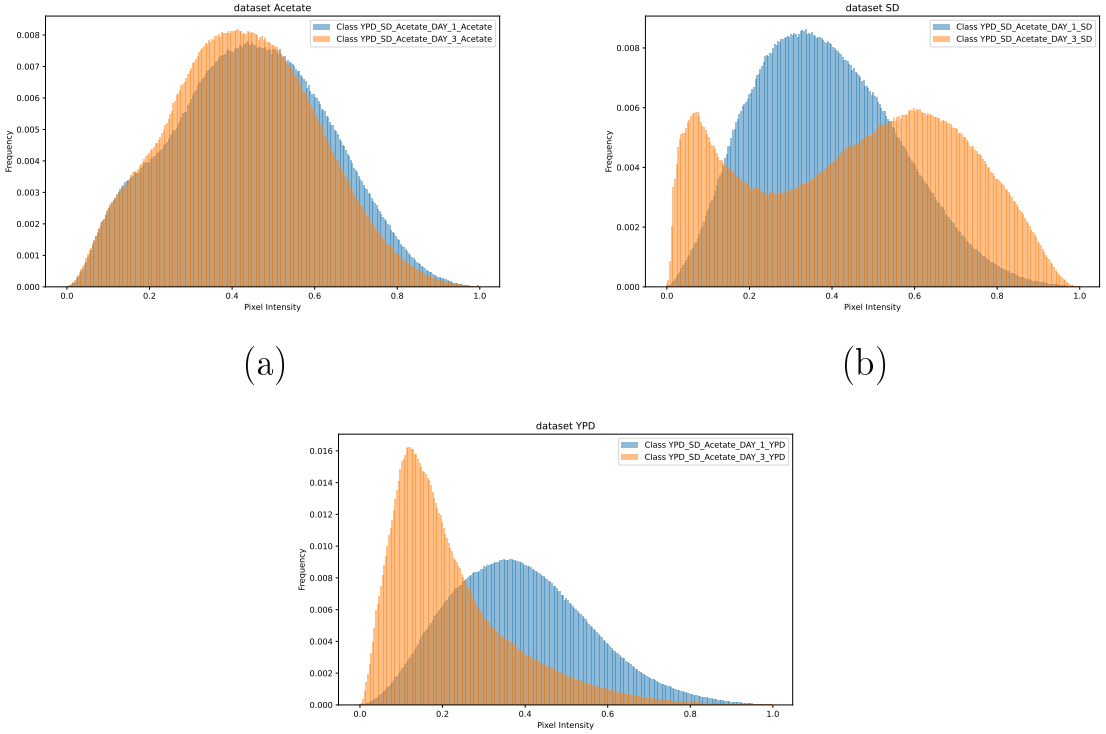


Figure 4.1: (a) Intensity histogram comparison between mitochondria from day 1 and day 3 in Acetate enviroment. (b) Intensity histogram comparison between mitochondria from day 1 and day 3 in SD enviroment. (c) Intensity histogram comparison between mitochondria from day 1 and day 3 in YPD enviroment.

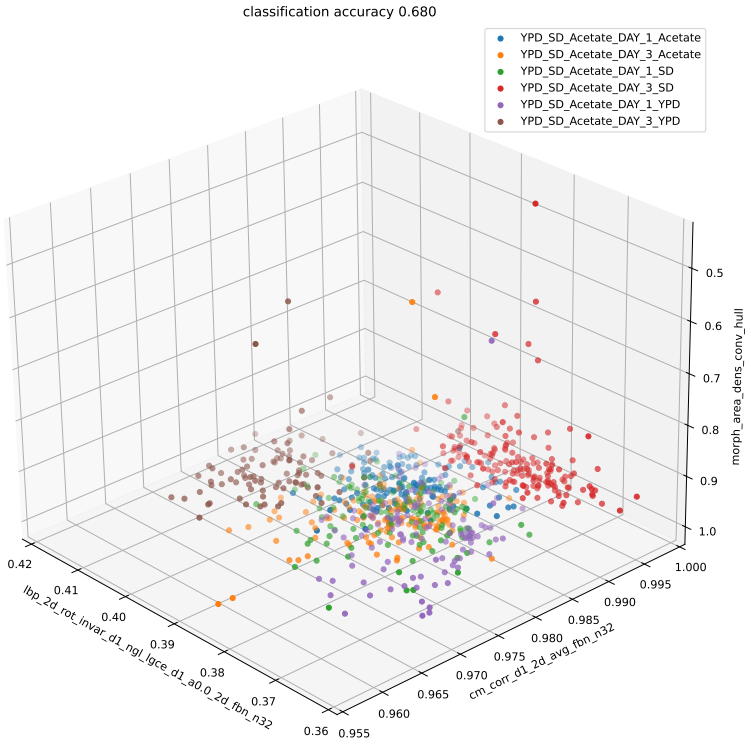


Figure 4.2: Example of classification individucl mitochondria into six categories by top three features selected.

Conclusion

Bibliography

[1] I.M.G.M. Hemel, B.P.H. Engelen, N. Luber, and M. Gerards. A hitchhiker's guide to mitochondrial quantification. *Mitochondrion*, 59:216–224, July 2021.

[2] Anne Humeau-Heurtier. Texture feature extraction methods: A survey. *IEEE Access*, 7:8975–9000, 2019.

[3] Alex Zwanenburg and Steffen Löck. Mirp: A python package for standardised radiomics. *Journal of Open Source Software*, 9(99):6413, 2024.

[4] Alex Zwanenburg, Martin Vallières, Mahmoud A. Abdalah, Hugo J. W. L. Aerts, Vincent Andrearczyk, Aditya Apte, Saeed Ashrafinia, Spyridon Bakas, Roelof J. Beukinga, Ronald Boellaard, Marta Bogowicz, Luca Boldrini, Irène Buvat, Gary J. R. Cook, Christos Davatzikos, Adrien Depeursinge, Marie-Charlotte Desseroit, Nicola Dinapoli, Cuong Viet Dinh, Sebastian Echegaray, Issam El Naqa, Andriy Y. Fedorov, Roberto Gatta, Robert J. Gillies, Vicky Goh, Michael Götz, Matthias Guckenberger, Sung Min Ha, Mathieu Hatt, Fabian Isensee, Philippe Lambin, Stefan Leger, Ralph T.H. Leijenaar, Jacopo Lenkowicz, Fiona Lippert, Are Losnegård, Klaus H. Maier-Hein, Olivier Morin, Henning Müller, Sandy Napel, Christophe Nioche, Fanny Orlhac, Sarthak Pati, Elisabeth A.G. Pfaehler, Arman Rahmim, Arvind U.K. Rao, Jonas Scherer, Muhammad Musib Siddique, Nanna M. Sijtsema, Jairo Socarras Fernandez, Emiliano Spezi, Roel J.H.M. Steenbakkers, Stephanie Tanadini-Lang, Daniela Thorwarth, Esther G.C. Troost, Taman Upadhaya, Vincenzo Valentini, Lisanne V. van Dijk, Joost van Griethuysen, Floris H.P. van Velden, Philip Whybra, Christian Richter, and Steffen Löck. The image biomarker standardization initiative: Standardized quantitative radiomics for high-throughput image-based phenotyping. *Radiology*, 295(2):328–338, May 2020.