University of Warsaw

Faculty of Mathematics, Informatics and Mechanics

Łukasz Rączkowski

Student no. 370887

Computational methods for analysis of the tumor microenvironment in histopathological images

PhD dissertation

in COMPUTER SCIENCE

Supervisor:

Dr hab. Ewa Szczurek

Institute of Informatics

Warsaw, March 2022

Abstract

The field of computer vision developed a range of low-level image features over the years, which try to encode semantic content of digital images. These features can be used in many applications, one of which is content-based image retrieval. Finding similar images requires the features to capture many aspects of image content and thus it is a quite challenging task. Another application for image features is image classification. Thanks to advancements in deep learning, it is now possible to eschew manual feature engineering altogether and generate image features from raw image data. This development made image classification possible in many previously challenging fields, such as digital pathology. Automatic classification of tissue types in histopathological slides has enormous potential to advance cancer research, as it allows to quantify the tumor microenvironment.

In this work, we explore application of informative image features in several domains, with the main focus put on digital pathology. We show that traditional low-level image features can be used to build a content-based image retrieval system, but are limited by their lack of semantic capacity. Next, we present an accurate, reliable and active framework for classification of histopathological tissue patches, which utilizes a Bayesian deep learning model, ARA-CNN. We study several aspects connected to uncertainty estimation in such models, exploring active learning and label noise detection. We then present how deep learning-based segmentation of histopathological slides can be used to introduce novel spatial features for tumor microenvironment quantification. We show that these image-based features are capable of predicting patient survival and classifying gene mutations in lung cancer, and are human-interpretable at the same time.

In summary, this work is a step forward in studying the tumor microenvironment and has the potential to be utilized in medical practice by pathologists.

Keywords

image feature, content-based image retrieval, machine learning, deep learning, digital pathology, histopathology, tumor microenvironment, colorectal cancer, lung cancer

Thesis domain (Socrates-Erasmus subject area codes)

11.3 Informatics, Computer Science

11.4 Artificial Intelligence

Subject classification

Applied computing → Life and medical sciences → Computational biology → Imaging

Computing methodologies

- → Machine learning
  - → Learning paradigms → Supervised learning → Supervised learning by classification
  - → Learning settings → Active learning settings
  - → Machine learning approaches → Neural networks
- → Artificial intelligence → Computer vision → Computer vision problems → Image segmentation

Tytuł pracy w języku polskim

Metody obliczeniowe do analizy mikrośrodowiska rakowego w obrazach histopatologicznych

Acknowledgments

First, I'd like to thank my supervisor, Dr hab. Ewa Szczurek, for her continued support over the years and for always believing in me. This dissertation was possible thanks to her many helpful efforts.

In addition, I'd like to thank my collaborators with whom I worked on my projects: Dr Anna Wróblewska from the Warsaw University of Technology, Dr Joanna Zambonelli from the Medical University of Warsaw, Dr Iwona Paśnik, Dr Marcin Nicoś, Dr Tomasz Kucharczyk, Prof. Justyna Szumiło and Prof. Paweł Krawczyk from the Medical University of Lublin, Dr Nicola Crosetto from the Karolinska Institutet, Dr Magdalena Budzinska, as well as Master's students Marcin Możejko (now PhD student) and Michał Kukiełka.

Finally, I'd like to thank my family and friends for supporting me along this long and perilous journey.

This study was supported by the research grant
2019/33/B/NZ2/00956
from the National Science Centre, Poland.

The work presented in this dissertation has been published in the following research papers and preprints:

Wróblewska, A. & Rączkowski, Ł. Visual Recommendation Use Case for an Online Marketplace Platform: *allegro.pl*. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, SIGIR '16, 591–594 (Association for Computing Machinery, New York, NY, USA, 2016). https://doi.org/10.1145/2911451.2926722.

Rączkowski, Ł., Możejko, M., Zambonelli, J. & Szczurek, E. ARA: accurate, reliable and active histopathological image classification framework with Bayesian deep learning. *Scientific Reports* 9, 14347 (2019). https://www.nature.com/articles/s41598-019-50587-1.

Rączkowski, Ł., Paśnik, I., Kukiełka, M., Nicoś, M., Budzinska, M.A., Kucharczyk, T., Szumiło, J., Krawczyk, P., Crosetto, N., Szczurek, E. Deep learning-based tumor microenvironment segmentation is predictive of tumor mutations and patient survival in non-small-cell lung cancer. Tech. Rep. (2021). https://www.biorxiv.org/content/10.1101/2021.10.09.462574v1.

Additional publications:

Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D.J., Hicks, S.C., Robinson, M.D., Vallejos, C.A., Beerenwinkel, N., Campbell, K.R., Mahfouz, A., Pinello, L., Skums, P., Stamatakis, A., Stephan-Otto Attolini, C., Aparicio, S., Baaijens, J., Balvert, M., de Barbanson, B., Cappuccio, A., Corleone, G., Dutilh, B.E., Florescu, M., Guryev, V., Holmer, R., Jahn, K., Jessurun Lobo, T., Keizer, E.M., Khatri, I., Kiełbasa, S.M., Korbel, J.O., Kozlov, A.M., Kuo, T., Lelieveldt, B.P., Mandoiu, I.I., Marioni, J.C., Marschall, T., Mölder, F., Niknejad, A., Rączkowski, Ł., Reinders, M., de Ridder, J., Saliba, A., Somarakis, A., Stegle, O., Theis, F.J., Yang, H., Zelikovsky, A., McHardy, A.C., Raphael, B.J., Shah, S.P., Schönhuth, A. Eleven grand challenges in single-cell data science. Genome Biol 21, 31 (2020). https://doi.org/10.1186/s13059-020-1926-6

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Digital image processing and computer vision systems have a long and storied history. Their building block was the neurophysiological research of Hubel and Wiesel conducted on cats in 1950s and 1960s [1, 2]. They discovered the existence of unique responses to different shapes in the visual cortex, which led them to conclude that vision is hierarchical. Along the visual pathway, neurons first recognize simple image characteristics such as edges or colors, followed by more advanced concepts like shapes, and then ultimately they feed into more abstract visual representations. This was expanded upon by David Marr in his work [3], which married biological aspects of vision with computer vision. He divided visual perception into independent modules: primal sketch, 2.5D sketch and 3D model representation. The primal sketch captures the spatial organization of perceived intensities in an image by encoding them with a set of low-level features corresponding to edges, bars, ends and blobs, represented by a 5-tuple (*type*, *position*, *orientation*, *scale*, *contrast*). The 2.5 sketch represents orientation and depth of observed surfaces, while the 3D representation is hierarchically organized in terms of surface and volumetric primitives. This framework worked as a baseline in computer vision for many years, with the primal sketch being an especially important part. The idea that an image can be described by a set of low-level features proved to be essential in building intelligent image analysis systems.

In recent years, mass access to digitized images sparked intense research on image processing, which in turn led to many new methods in computer vision [4]. These methodological improvements led to advancements across a variety of fields, such as medical imaging [5], autonomous driving [6], vehicle re-identification [7], image synthesis [8], and many more. One of the most challenging facets of the field is the extraction of semantic information from images,

i.e. understanding the contents within them. It is known as a semantic gap (**Figure 1.1**). It is not obvious how to bridge that gap between low-level features (colors, patterns, texture, etc.) and high-level semantic concepts (objects, categories). For many years this problem was a major roadblock on the way to building effective vision systems, as even simple tasks such as optical character recognition or object detection required that bridging step. However, with the advancement of machine learning techniques in this area, semantic information retrieval from images has become a reality.

One of the areas that faced the problem of the semantic gap was image matching, specifically what is known as content-based image retrieval (CBIR). It is a problem where similar images are retrieved from a large image database based on a query containing some input image. The prevailing approach to realize this was to extract low-level image features from all images in some database, combine them into feature vectors and index them in a high-dimensional feature space. Then the features from the input image were extracted as well and, based on some distance metric, most similar images in the database could be found. The problem with this approach is that the performance of the retrieval is highly dependent on what low-level features were used to construct feature vectors. For example, extracting only color information would not give good results when searching for similar dogs, because dogs of different breeds could have similar color. Thus, a lot of research was done to develop low-level features that would be as informative as possible, in order to successfully bridge the semantic gap [9]. However, ultimately these so-called manually engineered features proved to be inadequate and were superseded by deep learning methods [10].

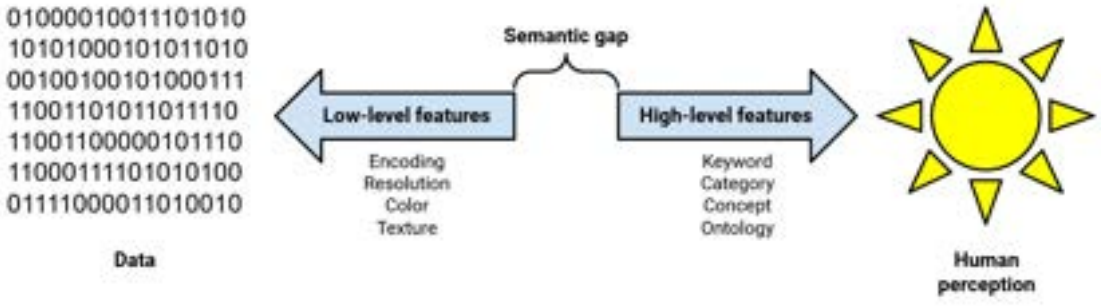

Figure 1.1: Semantic gap between low-level and high-level image features

Specifically, Convolutional Neural Networks (CNNs), which became the mainstream machine learning approach in image analysis thanks to the success of the groundbreaking AlexNet architecture [11], made a quantum leap on that front. The main cause of the effectiveness of

such models is their built-in ability to learn both low-level (in the form of convolutional filters) and high-level semantic image features automatically from the data, without any human supervision. This harkens back to the aforementioned research on the visual pathway. As such, the structure of CNNs is based on the biology of the visual cortex. Because of this, CNNs completely replaced previously used pipelines of traditional manually engineered feature representations. Another advantage is the fact that CNNs are very generalizable, because the learned features can be reused for different datasets and image processing tasks [12, 13]. This was not the case for the traditional approach, as it required the feature engineering step to happen every time for each new dataset or task.

Since AlexNet proved the effectiveness of deep learning compared to traditional machine learning methods, deep learning experienced a rapid growth. A wide range of architectures and accompanying methodological improvements was developed. CNNs were extended with residual connections in the ResNet architecture [14], which helped with the vanishing gradient problem [15]. In natural language processing, recurrent neural networks (RNNs) in the form of long short-term memory (LSTM) models or architectures with gated recurrent units (GRUs) proved to be very effective [16, 17, 18]. In addition to architectural developments, techniques such as batch normalization [19] and dropout [20] proved to be essential in increasing the performance of deep learning models. The field of deep learning research is still very active and it will remain like that for a long time to come.

1.1. Research directions undertaken in this dissertation

1.1.1. Uncertainty of deep learning models

One of these active research areas in deep learning is the study of uncertainty. Typically, a neural network that performs a classification task has a softmax activation function in its last layer, which normalizes the output to a probability distribution over predicted output classes. Such an output shows for each considered class the probability that a given input sample belongs to that class. However, depending on the training data and whether or not input samples are out of domain, such results might be misleading. Furthermore, it is possible to find such examples on purpose, in the form of so-called adversarial examples [21]. That is why it is advantageous to introduce the concept of uncertainty, which can be estimated for deep learning models in a Bayesian setting [22]. Thanks to techniques such as dropout and variational inference [23, 24] it is possible to measure uncertainty for each prediction

in addition to the standard output probabilities. Several measures of uncertainty have been proposed, such as expected entropy [25], mutual information [26] or softmax variance [27]. Each of them has its own uses, as they respond differently to different types of uncertainty [28]. Thus, it is interesting to study how these measures compare for a given dataset, deep learning architecture and application.

1.1.2. Label noise detection

A related research area is the problem of label noise. In supervised machine learning, the quality of the training data is of utmost importance, as it establishes the ground truth for the model to learn. If there are wrongly labeled data points in the training set, the model has no choice but to introduce such errors into its knowledge. That is why identifying such samples and removing them from the training data is a fundamental problem in supervised learning [29]. This is especially important in high-risk fields such as medicine, where models need to be as certain as possible, but at the same time it is somewhat expected that human annotators can make mistakes. There exist several approaches to dealing with this label noise problem, such as using metric functions [30], adding a de-noising layer to the network [31], finding outliers in a latent space trained with an autoencoder [32], or using uncertainty measures [33]. That last one is especially interesting, as it can be utilized for any network with variational inference. Because of that, the study of Bayesian deep networks in the context of label noise is a worthwhile avenue of research.

1.1.3. Active learning

Another related topic pertains to active learning. To train supervised learning models, annotated training data is needed. However, this requirement creates a limitation, as the annotation process can be costly and time-consuming. That is why optimizing the annotation time for new training datasets is an important aspect of deploying machine learning solutions in the real world. One technique that can be used is active learning [34]. It is an iterative procedure that first trains a given model with a small amount of annotated training data. Then, new samples are selected from a pool of unlabeled data according to some acquisition function. These samples are labeled by human annotators and appended to the training dataset. The model is retrained with this new data and the whole process repeats. The procedure ends when an acceptable level of performance is reached by the model. The main advantage of

this technique is that usually the number of annotated samples needed to reach that level of performance is much lower than it would have been with random sample selection. One way of defining the acquisition function in active learning is based on uncertainty. By selecting the most uncertain samples first, the model starts from learning the hardest examples, which is quite intuitive in its simplicity. There have been attempts at incorporating active learning for deep learning models with uncertainty serving as an acquisition function [35], but it is still an active research area.

1.1.4. Spatial metrics

In addition to working with image features, semantic information can be extracted from images in other ways. One of them concerns images with some distinct spatial organization, such as maps, satellite images or medical images (X-ray scans, MRI scans, histopathological tissue slides). Such images can be quantified by so-called spatial metrics, which try to summarize spatial organization encoded within. For example, such a metric can measure heterogeneity of animal species occupying a certain area. There have been many of these metrics proposed for application in various domains, from ecology [36, 37], through agriculture [38] to digital pathology [39, 40]. Development of new spatial metrics is a relevant and active field of research.

1.2. Applied research performed in this work and its motivations

The strategy of extracting informative data from images is applicable across many fields. Below I describe the areas of applied research which I explored in this dissertation, indicating critical issues that need to be tackled in these areas. This includes product images in e-commerce, but the main focus is put on the analysis of the tumor microenvironment in histopathological images.

1.2.1. Low-level image features for content-based image retrieval

CBIR systems can be used for what is known commonly as "search by image". This method of search has gained a lot of traction in recent years. More and more people use camera-equipped mobile phones and as such they always have a device capable of taking a digital photo, which could be used to search for similar objects. A recent survey [41] showed that 62% of Millenials and Gen Z consumers wish for visual search over any other new technology. This method of searching has many advantages over regular text-based queries, especially in domains such as

fashion. It is much easier to take a photo than to describe in detail a certain pattern or shape of a given piece of clothing. Several prominent companies have already deployed their own image retrieval solutions. The most well-known and most visited is Google's Search by Image [42]. Microsoft deployed its own visual search solution in Bing [43], built with scalability in mind. Yahoo built a visual similarity-based interactive search system, which led to more refined product recommendations [44]. Pinterest developed an image search platform, showing that content recommendation powered by visual search improves user engagement [45]. At eBay it was proven that image-based information can be used to quantify image similarity, which can be used to discern products with different visual appearances [46, 47]. Finally, Alibaba built a large-scale image search solution, which is heavily utilised by its consumers [48]. Thus, the critical issue to be tackled in this area is **the development of visual search or recommendation systems that utilize low-level image features**.

1.2.2. Bayesian CNN applied to tissue classification, with uncertainty used for active learning and identification of mislabeled samples

Machine learning for digital pathology

One of the most common applications of CNNs is image classification. Given a set of labels, each input image is categorized as one of these labels. Classification can be applied in many areas, but one of particular importance is digital pathology. It is a field of computer-aided analysis of digitized tissue samples taken from patients in order to help with diagnosis of various diseases, with cancer being the most prominent one. Doctors known as pathologists routinely inspect such images for cancer type identification and prognosis, and hematoxylin-eosin (H&E) stained slides have been used by them for over a hundred years. With such long history and proven applicability, histopathological imaging is expected to stay in common clinical practice for years to come [49]. However, the process of identifying and labeling tissue types is very time consuming, so pathologists are in short supply. That is why **new machine learning approaches are crucially needed to support pathologists in the analysis of H&E images**.

Furthermore, because digital pathology is medicine-related and works with histopathological tissue slides, models that process such data need to be as accurate as possible. If implemented in a clinical setting, they could be used for patient diagnosis or to decide upon a specific treatment, so there is no room for error. Multiple approaches to image classifica-

tion in digital pathology have already been established [50, 51], promising to aid the effort of pathologists in interpreting H&E images [52]. This process gained even more momentum after recent advancements in deep learning, sparking the creation of many different new models [53, 54, 55, 56, 57, 58, 59, 60]. However, the field of deep learning evolves quickly, and each new advancement has a potential to create better performing models, which in turn can open new avenues for clinical applications. Thus, **development of new, more accurate, tissue classification models is of crucial importance in digital pathology**.

Uncertainty estimation in the context of digital pathology

In addition to being accurate, machine learning models in digital pathology also need to be reliable. This means that it should be possible to measure the uncertainty of each prediction in order to ascertain if it can be trusted in a clinical setting. Various uncertainty estimation methods have already been proposed for use in digital pathology [61, 62, 63], but each of them has its advantages and drawbacks [63], and each needs to be evaluated anew for different datasets and model architectures. This means that **uncertainty estimation for deep learning models in the context of digital pathology is an important issue that needs to be tackled**.

Active learning in digital pathology

Machine learning models in general, and deep learning ones in particular, require vast amounts of training data to train accurately. Publicly available training datasets with annotated H&E images such as the Breast Cancer Histopathological Database [64] or the colorectal cancer patch dataset by *Kather et al.* [65] allow algorithm benchmarking and evaluation, which causes new method developments [66, 67]. However, generation of such datasets requires laborious workload of pathologists who process H&E images and assign labels to selected image regions. The requirement of meticulous pathological annotation limits the amount of data available for model training. As such, utilizing active learning in the field of digital pathology is definitely worthwhile. There have been multiple attempts at active learning for histopathological image classification in the literature, using both traditional machine learning [68, 69, 70, 71, 72] and deep learning [73, 74, 75, 76]. As such, it is clear that **creating a pathology-oriented framework that combines active learning with uncertainty measurement and produces accurate classification results** is a critical issue in this

field.

Identification of mislabeled training samples in digital pathology

Moreover, the problem of label noise is particularly important in digital pathology. If a training dataset has some cancer-related examples erroneously marked as non-cancer, then the trained model may miss a cancer diagnosis. That is why it is essential to have properly annotated data in this field. Unfortunately, recent developments in handling mislabeled data have been left largely unnoticed by the digital pathology community [77]. There are only a handful of works that tried to approach that problem for histopathological images [77, 78]. This means that the **problem of label noise in training data is hardly addressed in digital pathology and as such is an important issue to explore**.

1.2.3. Image-derived spatial metrics used as features for survival prediction and mutation classification in lung cancer

Importance of the tumor microenvironment

H&E image classification can be done at different scales of magnification - from pixel-level [79], through patch-based (where image slides are split into rectangular patches of small size) [58, 80] to larger regions of interest (large cutouts from the whole slide) [81]. The first two are de-facto also segmentation approaches - after classifying each subsection of the image, they can be merged back together to produce a segmentation of the whole slide. With slides segmented like that, it is possible to analyze the so-called tumor microenvironment (TME). H&E images portray the spatial architecture of the TME, including tumor cells, stromal cells, immune cells, and hypoxic/necrotic tissue areas and their reciprocal spatial arrangement (**Figure 1.2**). The TME plays an important role in cancer progression and metastasis, and thus it is critical to study its composition extensively [82]. Different tumors, even of the same type, have various genetic profiles resulting from gene mutations [83]. For a given cancer type, survival of individual patients can largely vary [84, 85]. Finally, the TME of different tumors is also different [86]. Thus, the **TME is very important in pathological examination and its study is a critical topic in cancer research**.

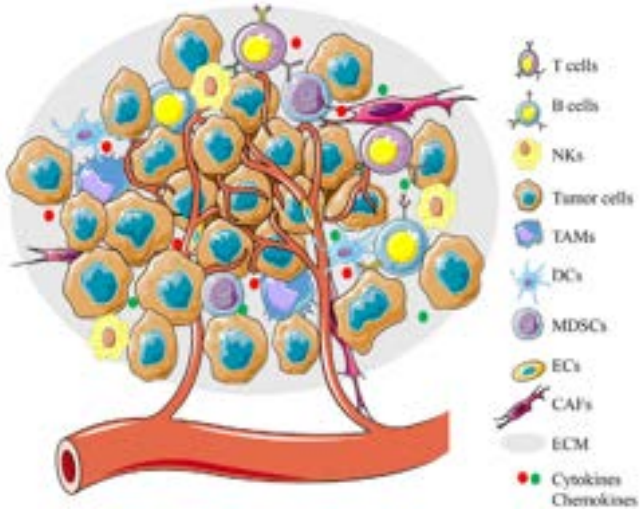

Figure 1.2: The tumor microenvironment (TME). The composition of the TME plays an important role in tumor development. Presence of different types of tissue around tumor cells has significant implications for prognosis and possible treatment options. Image source: Shi *et al.* [87]

Importance of lung cancer research

Lung cancer is the deadliest cancer type worldwide, and lung adenocarcinoma (LUAD) is becoming the mainly diagnosed subtype of lung cancer [88, 89]. LUAD includes a relatively higher proportion of cases without tobacco exposure. Thus, it has a more balanced molecular background and is more frequently associated with the presence of single somatic driver mutations that may be effectively managed with specific molecularly targeted therapies [90]. Several genes are known markers of response to treatment and survival in LUAD, including *EGFR*, *ALK*, *ROS1*, *BRAF*, *NTRK1-3*, *RET*, *MET*, *KRAS*, and diagnostic panels for targeted gene sequencing for detecting mutations in critical genes are routinely used in clinical practice [91]. Wide tumor spread, access to vessels, large areas of necrosis visible in H&E images, are associated with poor diagnosis [82], while abundance of immune cells indicates anti-tumor response of the immune system and associates with better survival [92, 93]. The TME also plays an important role in LUAD response to immunotherapy. Expression of PD-1 and PD-L1 on cancer or immune cells, as well as tumor mutation burden (TMB), are important biomarkers of immune checkpoint inhibitor efficiency [94, 95]. Due to these factors, **lung cancer study is of critical importance and should be especially focused on**.

Human-interpretable features in medical machine learning

Machine learning models used in clinical practice should in principle be explainable - clinicians should be able to interpret input features and draw conclusions based on their composition and importance. While deep learning-based models provide impressive performance when compared to traditional machine learning, their black-box nature often makes them unsuitable for real-world clinical deployment. In such cases, it is preferable to use human-interpretable features and explainable models. Thus, **the study and development of human-interpretable features is a critically important research area**.

Survival prediction in lung cancer

Computational prediction of patient survival from H&E images has been performed either based on spatial metrics or by using deep learning approaches. In the former case, spatial metrics are initially used to summarize the spatial arrangement of different tissues and next their correlation with survival is investigated. Alternatively, these spatial metrics are used as features in traditional machine learning algorithms [96, 97, 98, 99]. The TME can be very heterogeneous, so it is not obvious how to quantify it and what metrics to use. These metrics include proportion-based [39], clustering-based [100], and methods borrowed from ecology [40], and they have been applied to many different cancer types [97, 100, 101, 102, 103]. In general, all these metrics share a common trait, i.e., they incorporate only a limited number of tissue types at once, such as tumor cells and lymphocytes, tumor cells and stroma, etc. This approach cannot comprehensively capture the complexity of the TME. Thus, **there is an unmet need for an encompassing spatial metric that would consider many possible TME components at once**. In the case of approaches based on deep learning, deep neural networks are trained to predict patient survival directly from H&E images. Such methods are increasingly used for survival prediction and have been shown to perform comparably to or even better than spatial metric-based approaches [104, 105]. However, similarly to previously described applications in digital pathology, such deep learning methods for survival prediction lack explainability. Due to the complicated structure of these models and their number of parameters, it is not easy to surmise which parts of the TME are the most important for patient survival. As such, **creating explainable, i.e. human-interpretable, and TME-encompassing H&E image-based survival models is a critical issue**.

Mutation classification in lung cancer

Numerous methods for predicting gene mutations from H&E images were introduced and applied to a spectrum of cancers [106, 107, 108, 109, 110, 111], showing that such approaches can reveal links between the TME composition and mutations of selected genes. However, similarly to deep learning-based methods for patient survival, these models take raw image data as input and directly predict the presence of mutations. As such, it is hard to assess what parts of the TME are the most predictive of a given mutation. A recent study proved that human-interpretable features extracted from images segmented with deep learning methods can be successfully applied to predict phenotypic expression [112]. This suggests that the same approach can be implemented to predict gene mutations. **The issue of predicting gene mutations is of critical importance and utilizing human-interpretable features for that purpose has not been explored as of yet**.

Conditional Random Fields for tissue segmentation in H&E images

The effectiveness of the TME analysis based on spatial metrics depends on the quality of segmentations that are used to calculate them. The main issue with this lies in the fact that tissue classification models do not consider neighboring regions, thus running the risk of producing a lone, isolated region of patches classified as one tissue being surrounded by a region of patches classified as another tissue. They also produce very irregular borders between tissue regions, so it is important to smooth them out, especially for the purpose of calculating neighborhood-oriented spatial metrics. One method of mitigating this is to use a spatially-aware algorithm such as the Conditional Random Field (CRF) model [113]. This model has been utilized for the purpose of segmenting H&E images in the past, either as a post-processing step [114, 115, 116, 117, 118] or incorporated in a deep learning architecture and trained in an end-to-end manner [119, 120, 121]. The former approach is easier to integrate into existing solutions, as it does not require retraining the model. Consequently, **improving segmentation results is a critical problem and should be prioritized when new spatial metrics are designed**.

1.3. Contributions

This dissertation presents the following contributions:

1) Application of low-level image features in a large-scale CBIR system and its deployment within *allegro.pl*

2) New Bayesian deep learning architecture, ARA-CNN, within a new accurate, reliable and active tissue image classification framework, ARA

3) ARA-CNN model trained on a dataset of colorectal cancer patches from Kather *et al.* [65]

4) Study of uncertainty measures

5) Exploration of active learning in histopathology

6) Method for identification of mislabeled samples in digital pathology datasets

7) New dataset of LUAD tissue patches, *LubLung*

8) ARA-CNN model trained on the *LubLung* dataset

9) New dataset of segmented LUAD tissue slides from The Cancer Genome Atlas (TCGA), *SegLungTCGA*

10) New spatial metrics for the analysis of the TME, *tissue prevalence* (TIP) and *tumor microenvironment composition* (TMEC)

11) Application of these metrics in two tasks: survival prediction and mutation classification in LUAD

12) Application of the CRF model to segmented tissue slides in *LubLung*

1.4. Thesis contents

This dissertation is based on three publications that I worked on in the course of my PhD studies. In the first one, we presented a practical application of low-level image features in the context of visual product recommendations [122]. The second one introduced a new accurate, reliable and active deep learning framework for histopathological tissue classification [123]. The third one explored how deep learning-based tissue segmentation and human interpretable features can be used to predict patient survival and to classify gene mutations in lung cancer [124]. During my studies, I also contributed to a white paper on data science challenges in single cell sequencing analysis [125], which will not, however, be counted as part of the dissertation.

The dissertation is based on the above publications and describes how image features can be extracted from images and applied to several tasks. The first of these tasks is visual product recommendations (Chapter 2). The second task is tissue classification in colorectal cancer H&E tissue slides (Chapter 3). The third one pertains to tissue classification and segmentation in lung cancer, followed by the analysis of the TME in relation to patient survival and gene mutations, with the addition of utilizing the CRF model for improving tissue segmentation results (Chapter 4). The work is summed up in the discussion (Chapter 5).

Chapter 2

Low-level image features applied to similarity search in visual product recommendations

2.1. Visual product recommendations in e-commerce

E-commerce platforms have several ways of increasing user engagement in order to push their customers towards a purchase. One of these solutions is a recommendation system for suggesting other products that might be relevant to the user browsing a given product page. These recommended products can be chosen based on many criteria, such as previous purchases, browsing history, current trends, etc. One of these approaches is looking for products that are visually similar. This is particularly useful in product categories for which visual information is important or which contain products hard to describe with words, such as fashion, jewelry, paintings, etc.

One of such e-commerce platforms is *allegro.pl*. It is an online marketplace, where users can sell many different products and which offers a huge customer base - it is the most popular place to shop online in Poland. In addition to its recommendation system based on textual information, there arose a need for a visual recommendation system. To this end, we started a project with the goal of creating a CBIR system and deploying it in a production environment within a visually rich product category. In addition to this business-oriented goal, the project was also meant to research the usefulness of a set of low-level image features in the hard task of finding similar-looking products.

2.2. Analyzed data

2.2.1. Offer data on *allegro.pl*

Offers listed on *allegro.pl* are described by a limited set of well-defined attributes, a short title and a long description that contains unstructured data with a lot of additional marketing information and detailed product characteristics. Additionally, offers are placed in a hierarchical category structure. The structure begins with top-level categories (departments) such as 'Fashion', 'Electronics', 'Automotive', etc. These are followed by a series of nested, more detailed categories, ending with so-called leaf categories, which are very specific. Offers are placed only in these leaf categories. Each offer should also include at least one photo that shows the offered product. This photo is a so-called main offer image - it is shown in thumbnails on the offer listing and is displayed by default upon loading the offer page.

2.2.2. Quality of textual parameters

A large amount of offer parameters, especially in the 'Fashion' department, is unfortunately not very precise. For example, a 'dotted pattern' value for the 'Pattern' attribute may describe a large spectrum of dot sizes and configurations. Furthermore, traditional textual color tags have only around 10 very general values, e.g. 'shades of yellow' or 'shades of brown'. Despite the fact that color and pattern attributes are obligatory, *allegro.pl* sellers often specify them ambiguously as 'other color', 'other pattern', 'multicolor' or assign multiple values (e.g. both 'black' and 'white'). Thus, recommendations based only on text attributes can be quite vague and imprecise. A cursory data analysis in a single fashion category, 'Dresses' (**Figure 2.1**), shows that a large fraction of fashion items listed on *allegro.pl* lacks precise color information. Indeed, offers with the color attribute value set to 'multicolor' or 'other color' constitute 10.83% of all offers in that category. Furthermore, a significant 37.23% of offers for dresses has multiple values assigned to the color attribute. This means that only 51.94% of offers in the 'Dresses' category have clearly defined color information.

Such a huge disparity in attribute quality means that another solution must be used to effectively search for and recommend similar products. A great source of data for that purpose lies in product images. By calculating low-level image features, semantic information can be extracted from images, which can in turn be used to get a better understanding of *allegro.pl* offers.

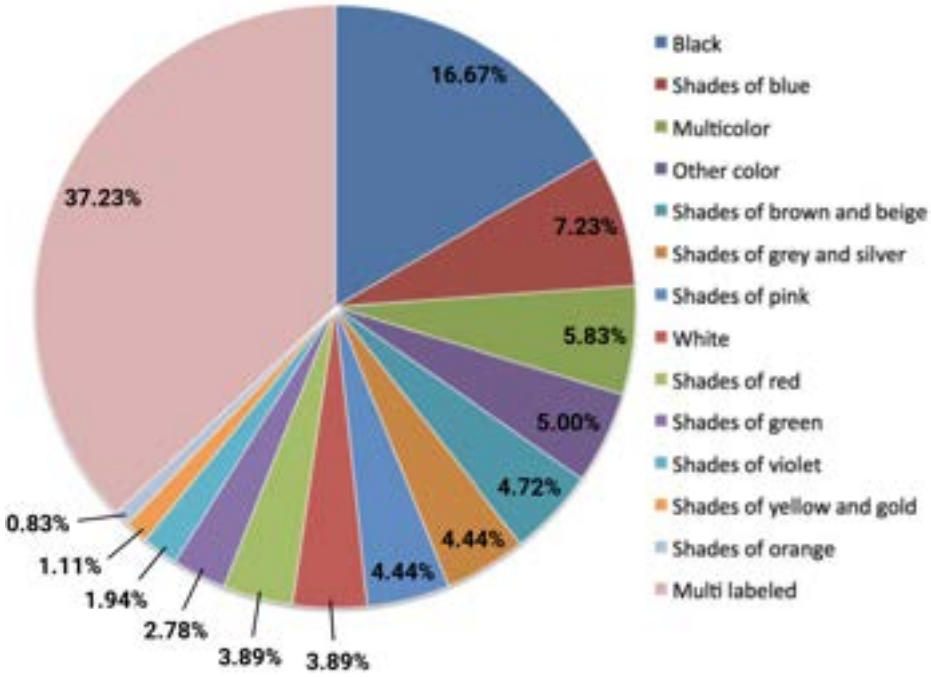

Figure 2.1: Color attribute distribution in the 'Dresses' category (data up to date as of March 2016)

2.2.3. Offer images

The quality of images attached to product offers varies greatly depending on the category. This is also the case in the 'Fashion' department. Some categories are mostly populated by professional sellers, who prepare better looking images in order to gain both attention and trust of customers, while others are a mix of different sellers who list offers with images of varying quality. An example of the latter is the 'Dresses' category, as it contains both clearly presented products with a single product model in the center of the picture (**Figure 2.2a**) and offers with images that contain several color options, different angles, or are not very clear (**Figure 2.2b**). To overcome this quality hurdle, we have decided to only work with images from a special part of *allegro.pl* called 'brand zone'. These images usually have a bright background and the main object/product is in the middle. Thanks to this we were able to easily crop the essential part of every image. Moreover, we used only the main images - these images are the ones that most customers look at, so it made sense to limit the search only to them.

Figure 2.2: Example images in the 'Dresses' category. **(a)** Clearly presented product images from the 'brand zone' section. The product is always in the center and the background is usually uniform. **(b)** Problematic images outside of the 'brand zone' section. They can contain several color options, different angles in a single photo or bad picture quality.

2.3. Tools and methods

2.3.1. Low-level image features

Before the production deployment, a series of experiments was performed in order to find which low-level image features work best for the use-case of finding similar-looking dresses. The following features were tested:

- auto color correlogram (ACC) [126]
- edge histogram descriptor (EHD) [127]
- binary patterns pyramid (BPP) [128, 129]
- joint histogram (JH) [129]
- palette power (PP) [46]

Auto color correlogram

The ACC feature captures spatial correlation between pixels of the same color. Let I be an image, which is quantized into m colors $c_1, \ldots, c_m$. For a pixel $p = (x, y) \in I$, where x and y are respectively horizontal and vertical coordinates, let $I(p)$ denote its color. Let $I_c = \{p \mid I(p) = c\}$. Then, the notation $p \in I_c$ is synonymous with $p \in I,\ I(p) = c$.

For a pair of pixels $p_1 = (x_1, y_1)$, $p_2 = (x_2, y_2)$, the distance between them is defined as $|p_1 - p_2| = max\{|x_1 - x_2|, |y_1 - y_2|\}$. Let a max distance d be fixed *a priori*. Then, the correlogram of I for colors c_i and c_j is defined as:

$$\gamma_{c_i,c_j}^{(k)}(I) = P(p_2 \in I_{c_j} \mid |p_1 - p_2| = k, p_1 \in I_{c_i}, p_2 \in I), \tag{2.1}$$

where $i, j \in \{1, \ldots, m\}$, $k \in \{1, \ldots, d\}$. The autocorrelogram (or ACC) of I for color c is then defined as follows:

$$\alpha_c^{(k)}(I) = \gamma_{c,c}^{(k)}(I) \tag{2.2}$$

As it needs to be computed for all $c \in \{c_1, \ldots, c_m\}$ and $k \in \{1, \ldots, d\}$, the final $\alpha(I)$ is an $m \times d$ matrix A. For the purpose of computing the ACC distance between images, A is converted to an md-dimensional vector row-wise.

Edge histogram descriptor

The EHD feature represents the combined distribution of 5 different types of edges in an image. An image I is divided into 16 sub-images of equal size, $I_{(i,j)}$, where $i, j \in \{0, 1, 2, 3\}$. Then, in each sub-image a histogram of edge type distribution is generated. The sub-image is split into non-overlapping image blocks and each of these blocks is tested for the presence of an edge. The 5 types of edges are: vertical, horizontal, 45-degree, 135-degree, non-directional. Histograms from all sub-images are combined into a single 80-element (16 sub-images times 5 edge types) vector that sums up the whole image I.

Binary patterns pyramid

The BPP feature captures the spatial distribution of edges, thus summing up the shapes within an image. First, it detects edges in the input image I with the Canny edge detector [130]. Then, it builds a hierarchy of progressively smaller windows from the grayscale representation of I - it divides I into 4 parts, then each part is again divided into 4 smaller parts (the full image I is treated as level 0). For each window, a histogram of rotation-invariant local binary patterns (LBP) [131, 132] is computed along pixels containing an edge. For every one of such pixels, an 8 bit vector is built, where each element represents a neighbor and is set to 1 if the grayscale value of the neighbor pixel is larger. These vectors are binned in a histogram. Finally, pattern histograms from each window at each subdivision level are appended together into a single histogram, which describes the whole I.

Joint histogram

The JH feature is a histogram of color distribution in RGB color space. For each pixel p in an image I, its color $I(p)$ is quantized into a new color c_i, $i \in \{1, \ldots, 64\}$:

$$c_i = \frac{B}{85} + 4 \cdot \frac{G}{85} + 16 \cdot \frac{R}{85}, \tag{2.3}$$

where R, G, B are the red, green and blue components of $I(p)$, respectively. Additionally, each p is assigned a rank from 0 to 8 based on the number of surrounding pixels for which the grayscale intensity is larger than the grayscale intensity of p. Then, the combined histogram of ranks for each c_i is computed, yielding a 576-element (64 colors times 9 ranks) vector.

Palette power

The PP feature is a histogram of color distribution in the HSV color space. In contrast to using RGB values, the HSV representation allows better handling of illumination variations, such as shadows or highlights. To calculate PP, the image I needs to be converted into the HSV color space first. For each RGB pixel p, the hue H, saturation S and value V channels are computed as such:

$$\begin{aligned} V &= \max(R, G, B) \\ S &= \begin{cases} \frac{V - \min(R,G,B)}{V} & \text{if } V > 0 \\ 0 & \text{otherwise} \end{cases} \\ H &= \begin{cases} \frac{60(G-B)}{S} & \text{if } V = R \\ 120 + \frac{60(B-R)}{S} & \text{if } V = G \\ 240 + \frac{60(R-G)}{S} & \text{if } V = B \end{cases} \end{aligned} \tag{2.4}$$

Then, for each of these channels plus for the grayscale channel G, a histogram is computed. There are 24 bins for the H channel, 8 for the S channel, 8 for the V channel and 8 for the G channel. These histograms are combined into a single 48-element vector, which constitutes the PP feature.

2.3.2. Distance measures

In order to compare image features for different images, a distance measure needs to be introduced. Then, for a given image, the distance to all indexed images can be computed and the closest ones can be returned as the most similar. In this project, three different distance measures were tested and evaluated.

Tanimoto distance

Given two N-dimensional histograms h and g, the Tanimoto distance [133] is defined as:

$$T(h,g) = 1 - \frac{\frac{\sum_i h_i g_i}{\sum_i h_i \sum_i g_i}}{\frac{\sum_i h_i^2}{(\sum_i h_i)^2} + \frac{\sum_i g_i^2}{(\sum_i g_i)^2} - \frac{\sum_i h_i g_i}{\sum_i h_i \sum_i g_i}}, \tag{2.5}$$

where all summations go from $i = 1$ to N.

Earth mover's distance

Given two N-dimensional histograms h and g, the Earth mover's distance [134] is defined as:

$$EMD(h,g) = \sum_{i=1}^{N}(|\sum_{j=1}^{i} h_j - \sum_{j=1}^{i} g_j|). \tag{2.6}$$

Hellinger distance

Given two N-dimensional histograms h and g, the Hellinger distance [135] is defined as:

$$H(h,g) = \sqrt{1 - \sum_{i=1}^{N} \sqrt{\frac{h_i g_i}{\sum_{j=1}^{N} h_j \sum_{j=1}^{N} g_j}}}. \tag{2.7}$$

2.3.3. Solution architecture

The following open-source projects were utilized in the development of the backend: Elasticsearch (ES) [136], Lucene Image Retrieval (LIRE) [129], OpenCV [137] and elasticsearch-image [138]. ES served as a database for image feature vectors and also as a search engine. The image features could be incorporated into an ES index thanks to the elasticsearch-image plugin and LIRE. The latter is a library that includes a set of low-level image features based on MPEG-7 edge detection and color histogram algorithms. From among those, we used ACC, EHD, BPP and JH. We extended LIRE with a new image feature, PP, and also added an image-cropping utility, which was possible thanks to OpenCV. We used OpenCV because

of its efficient algorithms and the potential of utilizing GPU (Graphics Processing Unit) computations.

The second part of the project was the development of a production-ready system. This required the creation of two additional components: an indexer service and a public API (**Figure 2.3**). The indexer service was a RESTful microservice subscribed as a client to Allegro's internal event queue. That queue was feeded with events whenever an offer was created, removed or changed. The indexer service received these events and was able to filter out unnecessary categories and offers outside of the 'brand zone', which was followed by adding the desired images into the ES index.

In addition to this, a public visual search API was developed in order to query the ES index from Allegro's frontend. The API was a RESTful microservice as well and exposed an HTTP endpoint which responded with a list of offers with images similar to the requested offer.

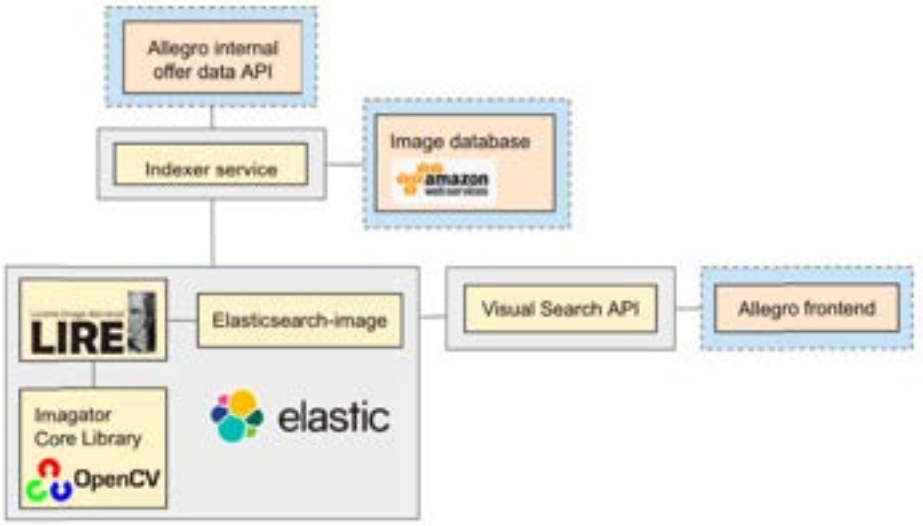

Figure 2.3: Architecture of the production-ready system. Gray background denotes backend components, while blue indicates external points of contact.

With these components, it was possible to create a working CBIR system. First, all main images from 'brand zone' offers in the 'Dresses' category were indexed in ES. For each image, a set of best performing low-level features was computed and put as a record in ES. Then, the indexer service was deployed, which kept the ES index synchronized with the state of the offers. After that, the visual search API and changes on the frontend could be deployed. On the offer page, whenever a customer clicked on the "Show similar" button, a request was

sent to the API, which queried the ES index. The index looked for records with the closest distance (according to the Tanimoto metric, as it was the best out of the three tested ones) to the given image and returned a list of 20 images, sorted in the order of ascending distance. The API then wrapped this response in necessary offer information and returned this to the frontend, which showed a pop-up gallery of similarly looking dresses (**Figure 2.4**).

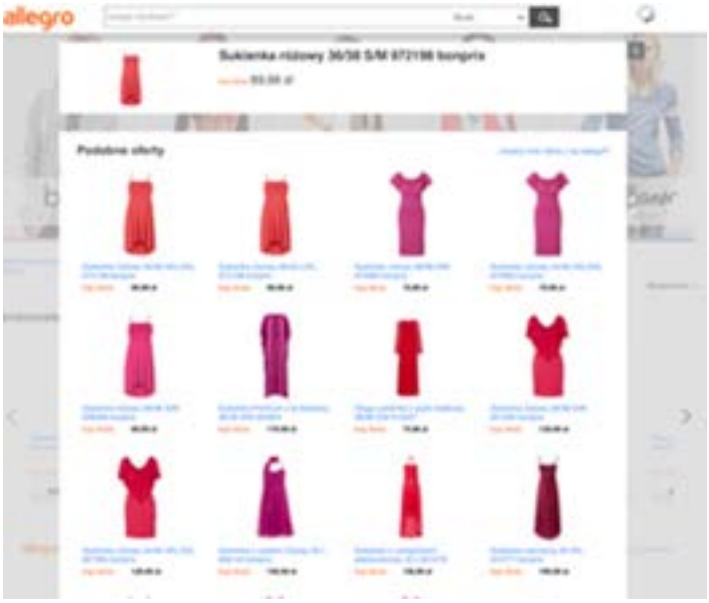

Figure 2.4: An example result with visually similar offers in the 'Dresses' category. For the image at the top, a list of 20 visually similar offers is returned, sorted by the similarity in descending order.

2.4. Results

2.4.1. Automatic tests

In order to asses the usefulness of different image features in the task of finding similar-looking dress images, we prepared a set of automated tests. The only way to automatically check the performance of image matching was to utilize the textual offer parameters. Three variants were chosen: only color, color + pattern, color + pattern + style. For a pair of offers to be counted as a match, all parameter values had to be the same in both of them. We uniformly sampled a set of 1300 dress offers and their images, after which for each image we performed visual search among all remaining images. The metric used to measure matching performance was mean average precision (MAP), defined as:

$$MAP@n = \frac{\sum_{q=1}^{Q} AveP@n(q)}{Q}, \tag{2.8}$$

where Q is the number of queries and $AveP@n$ is an average precision score. Specifically, $AveP@n$ for query q is defined as follows:

$$AveP@n(q) = \frac{\sum_{k=1}^{n} Pr(k)\ r(k)}{R} \tag{2.9}$$

where n is the number of retrieved images (e.g. for $AveP@5$, $n = 5$), k is the rank of a retrieved image, $Pr(k)$ is precision for the first k images, $r(k)$ is a function that grades an image 1 if it is a match and 0 otherwise, R is the number of good matches among the retrieved images.

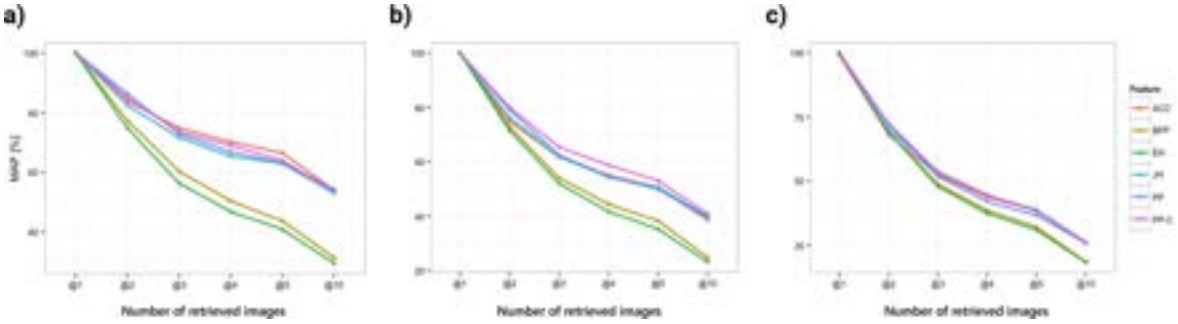

Figure 2.5: Automatic test results. **(a)** MAP results for the color parameter. **(b)** MAP results for the color and pattern parameters. **(C)** MAP results for the color, pattern and style parameters.

In the experiment where only the color parameter was considered, the best performing features were ACC, PP-C (center-cropped variant of PP), PP and JH, with BPP and EH falling behind significantly (**Figure 2.5a**) for every cutoff except @1, where all features scored a perfect 100%. The bigger the cutoff, the lower the score and the bigger the difference between the four leading features and BPP plus EH (up to ∼30% for MAP@10). This result is not surprising, because these two low-performing features extract edge information, which is not informative when it comes to finding dresses of matching color.

The experiment with the color + pattern parameters gave overall similar results, however in this case there was a clear best feature - PP-C (**Figure 2.5b**). It gave the best MAP across all cutoffs. PP, JH and ACC were slightly behind, while BPP and EH were again the worst performing features. Compared to the previous experiment, the gap between the worst features and the rest is different - it is ∼15% for MAP@10. It is caused by the drop in performance of PP-C, PP, JH and ACC - requiring the pattern parameter to match in

addition to color makes these metrics less informative. Conversely, BPP and EH didn't drop as much, meaning that edge-based features do indeed capture some pattern information.

The last experiment included the color + pattern + style parameters. In terms of results, they are again quite similar to previous experiments - PP-C is the best feature, followed closely by JH, ACC and PP (**Figure 2.5c**). BPP and EH are the worst performing features. This time the gap between the latter two and the rest is very small, ∼7%. The cause is very similar to the previous experiment - results for PP-C, PP, JH and ACC decreased much more than for BPP and EH. As such, edge-based features also capture some information about the style of dresses.

2.4.2. User tests

In addition to automatic tests, two rounds of tests with human users were performed. To this end, we prepared a simple web application that displayed an image of a dress and top five images indicated by the system as the most similar to it. Users were then asked to grade the similarity for each of these images on a scale from 1 to 5, where 1 meant no similarity and 5 meant that presented products were very similar. In total there were 10 users who performed this task. There were 66 main images which were used as input and the pool of searched images numbered 1166. The main images were sampled so that the distribution of their corresponding color and pattern parameters matched the distribution of these parameters in the 'Dresses' category within the 'brand zone'.

The results of the first user assessment round show that overall the suggested images were matched quite well (**Table 2.1**). In the first test phase the best-performing feature was ACC, with a mean user score of 3.26 ± 1.27. In this test users were not told what to focus on, they assessed overall similarity. For the ACC-C variant (center-cropped ACC), users were specifically asked to assess only color similarity. This resulted in a worse mean score, 2.82 ± 1.09, indicating that taking only the central region of an image for the ACC feature is not enough. The second best result was for the EH-C (center-cropped variant of EH) feature, with a mean user score of 2.90 ± 1.23. In this test users were told to focus only on pattern similarity. This result is significantly worse than for the general ACC assessment, which means that for general similarity, users mainly focused on colors instead of patterns. This is further supported by the results of the last test variant, where both ACC-C and EH-C were used at the same time and users were specifically asked to assess both color and pattern

Table 2.1: Results of user tests. ACC-C and EH-C indicate center-cropped versions of ACC and EH.

Test features	**User score**	**Score count**
1st test round		
ACC-C (users assessed only color similarity)	2.82 ± 1.09	990
ACC	3.26 ± 1.27	750
EH-C (users assessed only pattern similarity)	2.90 ± 1.23	990
ACC-C + EH-C (users assessed both color and pattern similarity)	2.80 ± 1.04	990
2nd test round		
JH	3.32 ± 1.05	330
PP	3.08 ± 1.44	465
JH + PP	3.05 ± 1.26	575
BPP + JH	2.41 ± 1.41	655

similarity - the mean score was 2.80 ± 1.04. Based on these results, the ACC feature was chosen for the initial production deployment.

The second round of user tests showed promising results as well. In all of these, users were not asked to focus on a specific aspect of images, so they judged overall similarity. The best performing feature was JH, which gave a mean user score of 3.32 ± 1.05. The next best one was PP, with a mean score of 3.08 ± 1.44. A combination of these two features, JH+PP, gave a score of 3.05 ± 1.26, so worse than individual variants. Finally, a combination of BPP+JH was also tested, but resulted in a low mean score of 2.41 ± 1.41. This again indicates that the BPP feature is not a good fit for the task of matching dress images, supporting the results of automatic tests.

2.5. Conclusions

Low-level image features can be successfully utilized for creating a production-ready image recommendation system. Thanks to readily available open-source tools it was possible to

build such a system from scratch and then test its performance, both with automatic and user tests. In the end, the system was deployed in a production environment on *allegro.pl* and served users in the 'Dresses' category.

Despite its effectiveness, the described approach to content-based image retrieval had several limitations. The first one was the issue of image quality - focusing only on the 'brand zone' section severely limited the pool of relevant offer images, but extending it was not possible due to the observed limitations of low-level image features. In general, it was clear that each of such features is very specific (quantifying mainly color, pattern, etc.) and cannot capture full semantic context by itself. Furthermore, combining color and pattern-oriented features proved to be ineffective in subjective user assessment, so clearly a different approach was needed. This led to my interest in deep learning methods and was a starting point for the next project.

Chapter 3

Histopathological tissue classification with Bayesian deep learning

3.1. The problem of tissue classification

In the field of digital pathology, the issue of tissue classification is of critical importance. Tissue identification in histologic slides is a crucial component of cancer diagnosis. Since pathologists' time is a limited resource, there is a need for automated methods of tissue classification that can help to share the load of these experts. Since the advent of deep learning, development of new algorithms for the purpose of tissue classification became an active research area. An especially important direction is the study of uncertainty, as models used for diagnostic purposes need to be as accurate and reliable as possible.

3.2. Deep learning concepts

3.2.1. Deep learning definition

Deep learning methods have revolutionized many fields of machine learning. What differentiates them from previous (now knows as traditional or shallow) approaches is twofold (**Figure 3.1**). Firstly, deep neural networks contain more hidden layers than standard multilayered perceptron models. This allows them to learn more complex abstractions. Secondly, deep learning models represent a current of machine learning research known as representation learning. This approach does not use manual feature engineering, but works with raw input data instead. So for images, raw pixel values are used, for sound raw waveforms, and for text

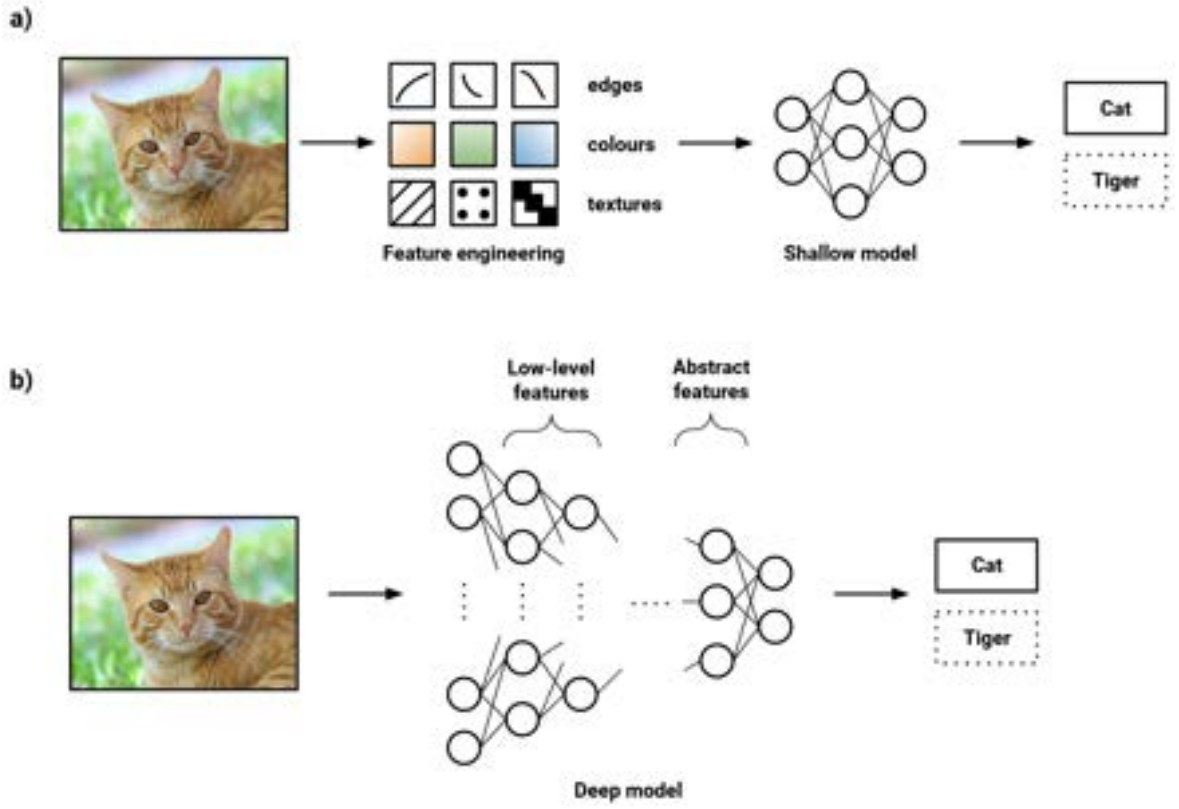

Figure 3.1: Traditional machine learning vs deep learning. **(a)** In traditional machine learning, raw input is transformed into a set of hand-engineered features, which are then used as input into a shallow model. If the model is a classifier, it then produces a classification into one of predefined output classes. **(b)** In deep learning, raw input is passed directly into a deep model. The model learns progressively more abstract representations, until it is able to differentiate between e.g. different animals.

it is just raw textual data. This method solves the selectivity-invariance dilemma that was a major roadblock for traditional models. Fundamentally, a feature extractor should both be selective to aspects of an image that are important for distinguishing different objects, but also invariant to aspects such as posing, viewing angle, lighting conditions, etc. When an image classifier is built, it should both be able to accurately capture differences between a tabby cat and a tiger when they are both viewed from the front as well as from other angles or when the former is in a jungle and the latter is in a living room. Traditional models struggled with that, even when features were designed specifically for the task at hand. Furthermore, feature engineering itself was a big problem, as it was often a time-consuming process and required expert knowledge to select proper features, as evidenced by the project described in Chapter 2.

By combining deep architectures with representation learning, deep learning is able to

automatically learn progressively more abstract features in each subsequent layer, ultimately being able to differentiate between concepts such as tiger and tabby cat.

3.2.2. Convolutional Neural Networks

CNNs are a type of deep neural networks designed specifically to work with data that has a grid-like structure. Thus, they are very well suited for images, which can be represented as 2D arrays of pixel values. The main element of these networks is a special kind of layer known as a convolutional layer. It utilizes a discrete convolution operation and is used to extract low-level image features, which are then combined and passed to deeper layers responsible for building more abstract representations. Specifically, edge, color and texture filters form motifs, motifs assemble into parts and parts form objects [139]. As such, CNNs build a semantic understanding of visual information that is put into them, similar to the visual cortex that their architecture is modeled on.

3.2.3. Convolution

The convolution operation s is in general defined as follows:

$$s(t) = (f * g)(t) = \int_{-\infty}^{\infty} f(a)g(t-a)da \tag{3.1}$$

where f and g are any real-valued functions. In order to apply this operation to images, the above integral needs to be discretized. For this, the t argument needs to be an integer, which allows to define s as:

$$s(t) = \sum_{a=-\infty}^{\infty} f(a)g(t-a) \tag{3.2}$$

Because in practical applications the values of f and g are stored in so-called tensors (multidimensional data arrays), the above sum can be implemented as a finite summation of non-zero tensor elements. In addition, in machine learning, the f function is known as input and for images can be treated as a function $I(i,j)$, which returns the pixel value for a pixel at position (i,j) in an image. Furthermore, the g function is known as the kernel or filter, which for images is a function $K_{(m,n)}(x,y)$, where $x \in \{1 \dots m\}$, $y \in \{1 \dots n\}$, m is the kernel width and n is the kernel height. It is worth noting that $m << W$, $n << H$, where W and H are, respectively, image width and height in pixels. In such a setup, the (commutatively flipped) convolution operation S for a pixel at position (i,j) is defined as:

$$S(i,j) = (I * K)(i,j) = \sum_{x=1}^{m} \sum_{y=1}^{n} I(i-x, j-y) K_{(m,n)}(x,y) \tag{3.3}$$

However, many deep learning libraries (including the Keras library [140] used to build the ARA-CNN model described later in this chapter) implement a similar function, cross-correlation, and treat it as if it was convolution. With this change, the kernel does not have to be flipped in relation to the input. Cross-correlation S' is defined as follows:

$$S'(i,j) = (I \star K)(i,j) = \sum_{x=1}^{m} \sum_{y=1}^{n} I(i+x, j+y) K_{(m,n)}(x,y) \tag{3.4}$$

There is one additional optimization that is often done - striding. Some positions of the input can be skipped, which essentially downsamples the output of S' and reduces the computational cost at the same time (**Figure 3.2**).

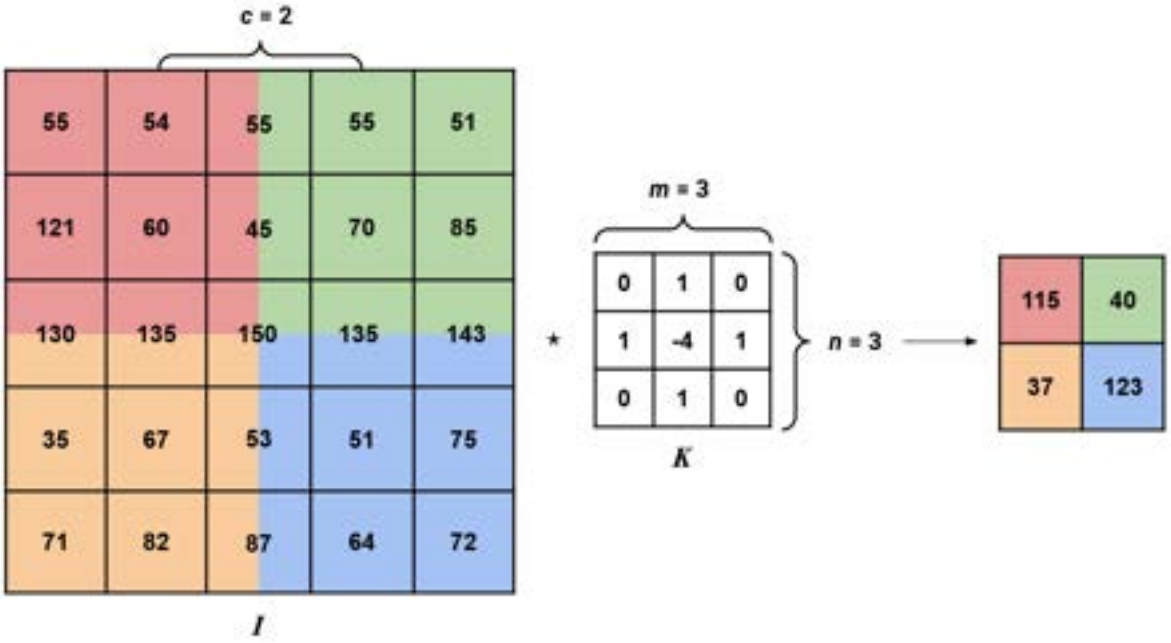

Figure 3.2: An example strided convolution operation. A 3×3 kernel K slides over a single-channel image I with a stride 2 - it skips 2 pixels both horizontally and vertically. Each submatrix of I bounded by K is multiplied element-wise by K and the results are summed, which leads to a downsampled result matrix - a so-called activation map.

In CNNs, the kernels are represented by neurons in convolutional layers. A single such layer with k neurons corresponds to k kernels, each of them with a common size. The weights for these kernels are trainable, so they are updated during neural network training. Upon convergence, these k kernels become in essence low-level image features responsible for extracting representations from input images.

3.2.4. Pooling

In image classification, it is important for the model to be invariant to small changes in the input. If an object in the input is slightly shifted or translated, the network should be able to still classify it properly. For this reason, a technique called pooling was introduced into CNN architectures [141].

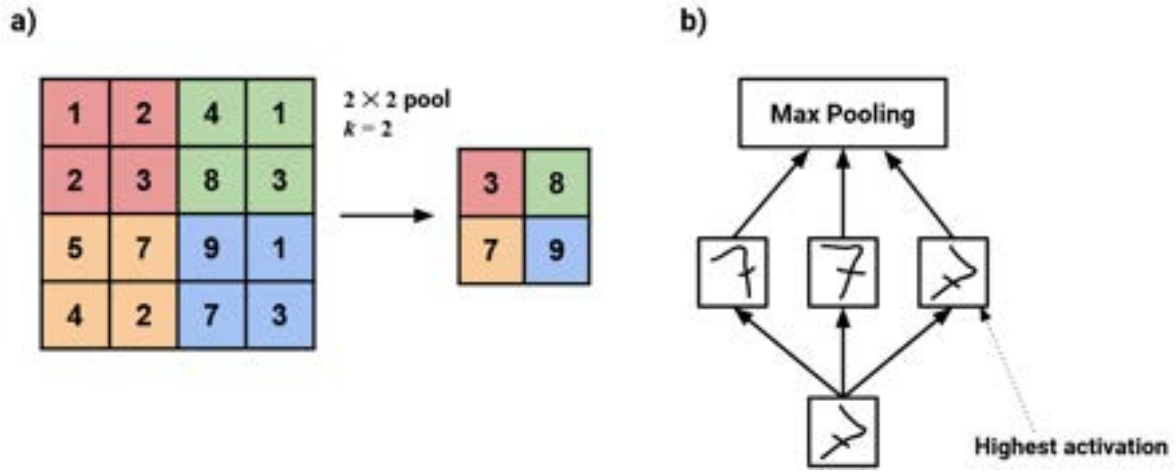

Figure 3.3: Max pooling. **(a)** A rectangular moving window slides over the output of the convolution layers and a maximum value from within it is selected. These maximum values are then propagated to the next part of the network. **(b)** Max pooling makes the model invariant to small translations and rotations in input images.

It is an aggregation operation applied to the output of convolution layers. The most common type of pooling is max pooling (**Figure 3.3a**). A rectangular windows slides over the aforementioned output, skipping over k pixels every time, and a maximum value is selected from within it. These maximum values are then propagated to the next part of the network. Skipping pixels reduces the number of network parameters and thus also the computational workload needed during both training and inference, as the next layer has approximately k times fewer inputs to process [142].

Max pooling makes the model invariant to small translations and rotations in input images (**Figure 3.3b**). The network learns filters from the data and the filter that is the closest match to the input image produces the highest activation value. Max pooling then chooses this value, thus propagating the correct signal further into the network.

3.2.5. Batch Normalization

When deep learning models are trained, the distribution of inputs for each layer changes as a result of modified parameters in preceding layers, which slows down the whole process [19]. This phenomenon is analogous to covariate shift [143], in which input distribution to a learning system changes in relation to previously seen data (e.g. different distributions for training and test data). However, this notion can be extended also to individual building blocks of the model. As such, in deep neural networks we can think of this as an internal covariate shift. A seminal technique introduced to combat this issue is Batch Normalization [19]. It works on the basis of normalizing layer inputs for each training mini-batch, which enables the use of higher learning rates and significantly speeds up the training. In other words, Batch Normalization is a method of adaptive reparametrization [142].

For a mini-batch B with m training samples and a single input x to a network unit, Batch Normalization performs the following operations:

$$\begin{aligned} \mu_B &= \frac{1}{m}\sum_{i=1}^{m} x_i \\ \sigma_B^2 &= \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_B)^2 \\ \hat{x}_i &= \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \\ y_i &= \gamma\hat{x}_i + \beta \end{aligned} \tag{3.5}$$

where μ_B is a mean for B, σ_B^2 is a variance for B, $\hat{x}_i$ is the normalized i-th sample in B, ϵ is a constant added for numerical stability and y_i is the result of a linear transform on $\hat{x}_i$, parameterized by learnable parameters γ and β.

This operation needs to be performed for all units in a layer, and can be be inserted after any hidden layer in the network. The two parameters, γ and β, reparameterize the model in a way that introduces both additive and multiplicative noise on the hidden units at training time. While the primary purpose of Batch Normalization is to improve optimization, this noise can have a regularizing effect as well.

3.2.6. Residual connections

Deep neural networks are inherently hard to train due to numerical limitations of gradient-based learning and backpropagation [144, 145]. Backpropagation is a learning algorithm

commonly used for neural network training, in which network weights are updated from the output to the input for each training example. As the depth of the model increases (i.e. the more layers are stacked), the vanishing gradient problem [15] becomes more pronounced. In very deep networks, the gradient calculated during backpropagation can become so small that the weights can no longer be updated and the training stops. Additionally, even if the training of such networks does converge, another problem can occur - degradation. As the depth increases, accuracy gets saturated and degrades rapidly. It has been shown that adding more layers leads to higher training error [14, 146, 147].

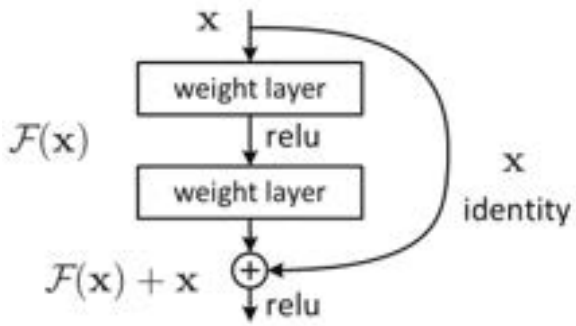

Figure 3.4: Residual connection. It is a shortcut identity connection, which allows the network to learn residual mappings. Image source: He *et al.* [14].

Both of these problems have been solved by the introduction of residual connections, and by extension, residual networks [14]. These connections allow the network to learn residual mappings via shortcut identity connections (**Figure 3.4**). Instead of learning $y = \mathcal{H}(x)$, where $\mathcal{H}$ is the original mapping, the network learns the following:

$$y = \mathcal{F}(x, W_i) + x, \tag{3.6}$$

where $\mathcal{F}$ is the residual mapping to be learned, x is the input vector to a residual block, y is the output vector and W_i are network weights for layer i. The residual mapping is easier to optimize than the original mapping, leading to faster training (residual connections do not introduce extra parameters) and lower error rates [14].

Interestingly, residual connections have their biological analogue as well. In the neocortex (part of the cerebral cortex), neurons in the sixth layer receive signal from neurons in the first layer, skipping all layers along the way [148]. It is still not yet clear how the brain utilizes these connections and if there is similarity to how they are used in artificial neural networks.

3.2.7. Loss function

During training, a loss function needs to be evaluated in order to calculate the error and use it to update network weights in the backpropagation procedure. In computer vision classification tasks, the most frequently used loss function is categorical cross-entropy.

For observation o, a set of M classes and class $y^* \in \{1 \dots M\}$, the probability of assigning the observation to that class is denoted as $P(y^*|o, \hat{\omega})$, where $\hat{\omega}$ represents the estimated parameters of the model. Categorical cross-entropy can then be defined as:

$$\mathcal{L}(\hat{\omega}; o; y^*) = -\sum_{y^*=1}^{M} \delta(y_o = y^*) \log(P(y^*|o, \hat{\omega})), \tag{3.7}$$

where δ is the Dirac function and y_o is the correct (i.e. ground truth) class for observation o.

3.2.8. Optimizer

For a neural network to learn, the loss function needs to be minimized with an optimization algorithm. The most used family of optimization algorithms for this purpose are gradient-based methods. For many years the stochastic gradient descent (SGD) [149, 150] algorithm was the most prominent one, however in recent years more methods have been developed and started to be widely used in the deep learning community, such as AdaGrad [151], RMSProp [152] and Adam [153]. These last three methods are in a subcategory of approaches that utilize adaptive learning rate.

The classic SGD algorithm updates network parameters $\hat{\omega}$ according to the following formula:

$$\hat{\omega} = \hat{\omega} - \eta \cdot \nabla_{\hat{\omega}} \mathcal{L}(\hat{\omega}; o; y^*) \tag{3.8}$$

where η is the learning rate parameter, $\mathcal{L}$ is the loss function, o is an observation from the training dataset and y^* is the class of that observation. Because the learning rate parameter is constant, the optimization procedure can overshoot minima and converge slowly or not converge at all. Adaptive learning rate methods counteract this by changing the learning rate dynamically during training. The algorithm used in this project was Adam, as it has been shown empirically to be stable and converge quicker than its peers [154].

3.2.9. Dropout

Due to their size, deep learning models are especially prone to overfitting - they can inadvertently learn from sampling noise instead of actual non-linearities in the training data. One

of the more popular and successful methods of combating this problem is dropout [20]. It works on the basis of randomly removing units in a neural network during training in order to simulate a committee of multiple different architectures.

Without dropout, the forward pass through a neural network can be expressed as:

$$\begin{aligned} x_i^{(l+1)} &= w_i^{(l+1)} \cdot y^{(l)} + b_i^{(l+1)}, \\ y_i^{(l+1)} &= f\left(x_i^{(l+1)}\right), \end{aligned} \tag{3.9}$$

where $\cdot$ denotes a dot product, $l \in 1, \ldots, L$ is the layer index, L is the number of network hidden layers, $x_i^{(l)}$ are the inputs into the i-th unit in layer l, $y^{(l)}$ are the outputs from layer l, $y_i^{(l)}$ is the output from the i-th unit in layer l, $w_i^{(l)}$ are the weights of connections coming into the i-th unit in layer l, $b_i^{(l)}$ is the bias for the i-th unit in layer l and f is an activation function.

With dropout, this changes so the forward pass becomes as such:

$$\begin{aligned} z^{(l)} &\sim \text{Bernoulli}(p), \\ \hat{y}^{(l)} &= z^{(l)} * y^{(l)}, \\ x_i^{(l+1)} &= w_i^{(l+1)} \cdot \hat{y}^{(l)} + b_i^{(l+1)}, \\ y_i^{(l+1)} &= f\left(x_i^{(l+1)}\right), \end{aligned} \tag{3.10}$$

where $z^{(l)}$ is a vector of independent Bernoulli random variables for layer l, each of which has a probability p of being set to 1, and $*$ is an element-wise product. In other words, each outgoing connection from layer l can be turned off with probability $1 - p$. This creates a thinned output $\hat{y}^{(l)}$. This thinning is done for each layer, which ultimately amounts to sampling a sub-network from a larger one [20]. As such, it can be seen as a method of stochastic regularization.

Dropout is traditionally enabled only during training. During inference it is turned off, but the weights are then scaled by a factor of p, so $\hat{\omega}_{test} = p\hat{\omega}$, where W is a set of weights from the trained model.

3.2.10. Variational dropout

In order to provide more accurate classification as well as uncertainty prediction, we adopted a popular method called variational dropout (also known as Monte Carlo dropout) [23]. The central idea of this technique is to keep dropout enabled by performing multiple model calls during prediction. Thanks to the fact that different units are dropped across different model

calls, it might be considered as Bayesian sampling from a variational distribution of models [155]. In a Bayesian setting, the parameters (i.e. weights) $\omega = (W_i)_{i=1}^{L}$ of a CNN model are treated as random variables, where W_i is a matrix of weights and L is the number of network layers. In variational inference, we approximate the intractable posterior distribution $P(\omega|D)$ by a simpler (variational) distribution $q(\omega)$, where D is the training dataset. The approximation results from dropout, as $q(\omega)$ is a distribution over such W_i for which columns are randomly set to 0 [23]. Thus, $q(\omega) = q(W_i)$ for every layer i can be defined as follows:

$$\begin{aligned} &z_{i,j} \sim \text{Bernoulli}(p_i) \text{ for } i = 1, \ldots, L,\ j = 1 \ldots, K_{i-1} \\ &W_i = M_i \cdot \text{diag}\left([z_{i,j}]_{j=1}^{K_i}\right), \end{aligned} \tag{3.11}$$

where $z_{i,j}$ are Bernoulli distributed random variables with probability p_i ($z_{i,j} = 0$ corresponds to unit j in layer $i-1$ being dropped as input to layer i), M_i are the variational parameters to be optimized (sampling from $q(\omega)$ is the same as performing dropout on the i-th layer in a network with weights $(M_i)_{i=1}^{L}$) and K_i is the number of units in the i-th layer. The diag operator maps vectors to diagonal matrices (with vector components placed on the diagonal).

Consequently, we assume that $\hat{\omega}_t \sim q(\omega)$, where $\hat{\omega}_t$ is an estimation of ω resulting from a variational dropout call t. With these assumptions, the following approximations can be derived [155]:

$$P(y^*|o, D) = \int P(y^*|o, \omega)P(\omega|D)d\omega \approx \int P(y^*|o, \omega)q(\omega)d\omega \approx \frac{1}{T}\sum_{t=1}^{T} P(y^*|o, \hat{\omega}_t) \tag{3.12}$$

where T is the number of variational samples. In our model we used $T = 50$.

3.2.11. Uncertainty of deep learning models

Variational dropout has enabled us to measure the uncertainty of predictions. We implemented two uncertainty measures: Entropy H and $BALD$ [156]. If the output of the model is a conditional probability distribution $P(y^*|o, D)$, then the measure H can be defined as the entropy of the predictive distribution:

$$H[P(y^*|o, D)] = -\sum_{y^* \in \{1 \ldots M\}} P(y^*|o, D) \log P(y^*|o, D) \tag{3.13}$$

The second uncertainty measure, $BALD$, is based on mutual information and measures the

information gain about the model parameters ω obtained from classifying observation o with label y^*. In the case of variational dropout, this can be expressed as the difference between entropy of the predictive distribution and the mean entropy of predictions across multiple model calls:

$$\begin{aligned} I(\omega, y^*|o, D) &= H[P(y^*|o, D)] - \mathbb{E}_{P(\omega|D)}[H[P(y^*|o, \omega)]] \\ &\simeq H[P(y^*|o, D)] - \frac{1}{T}\sum_{t=1}^{T} H[P(y^*|o, \hat{\omega}_t)] \end{aligned} \tag{3.14}$$

The difference between these two measures pertains to how they react to two different types of uncertainty in the data: epistemic and aleatory [28]. The former type is caused by a lack of knowledge - in terms of machine learning, this is analogous to a lack of data, so the posterior probability over model parameters is broad. The latter uncertainty is a result of noise in the data - no matter how much data the model has seen, if there is inherent noise then the best possible prediction may be highly uncertain [156]. In general, the H measure cannot distinguish these two types of uncertainty. If uncertainty of a new observation is measured by H, then the value would not depend on the underlying uncertainty type. On the other hand, it is believed that $BALD$ measures epistemic uncertainty of the model [156], so it would not return a high value if there is only aleatory uncertainty present. Depending on the dataset, one of these measures might work better than the other at catching and describing the uncertainty.

3.3. Analyzed data

3.3.1. H&E tissue slides

In clinical practice, in particular in oncology, taking tissue samples is a common procedure. To analyze the tumor type, malignancy and staging, biopsies or surgical resections are taken from the tumor. These tissue sections are then fixed against deterioration and the staining procedure follows. First, samples are treated with a solution containing one or more cationic aluminium-hematein metal complexes, commonly known as hematoxylin or (more strictly) hemalum. This is followed by differentiation in dilute acid alcohol and bluing in water. After that, samples are exposed to a solution of the anionic dye eosin Y [157, 158]. Hematoxylin reacts with cell nuclei, staining them purplish blue. Eosin acts as a counterstain, and there-

fore stains the extracellular matrix and cytoplasm with a contrasting pink. The hue of that pink varies depending on the extranuclear structure, giving a diverse overview of the whole tissue (**Figure 3.5**). Expert pathologists can analyze slides stained in this way. Among other features, they examine the pattern of cells, ratio of nuclei to cytoplasm and the density and pattern of the chromatin. They also study the structure, pattern and color of the cytoplasm [158]. This way, they can discern healthy tissue structures from pathologic ones, which allows them to accurately diagnose patients.

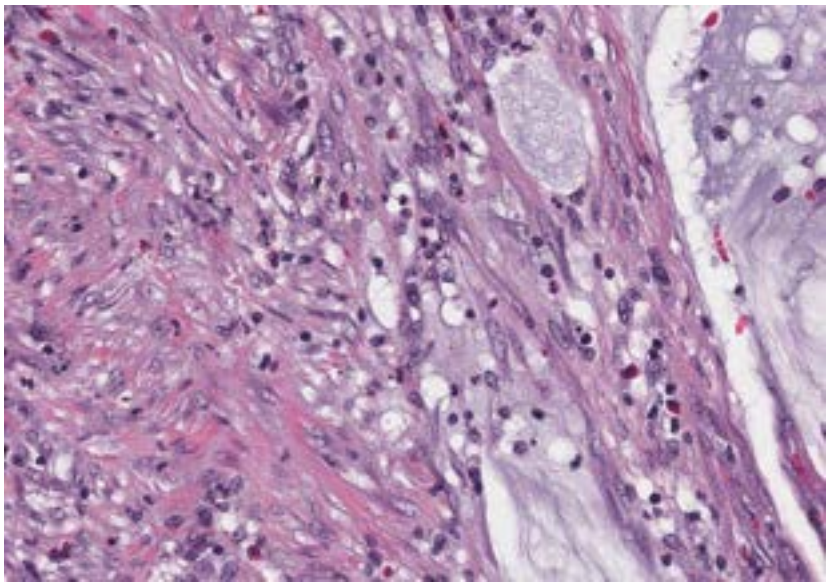

Figure 3.5: Example H&E tissue slide from the colon at high magnification (TCGA case ID TCGA-CM-5861). The dark blue areas are cell nuclei, while pink and violet regions are extranuclear structures.

3.3.2. Dataset of colorectal cancer tissue patches

The dataset analyzed in this project holds 5000 image patches belonging to eight balanced classes of histopathologically recognisable tissues [65]. The patches were pulled from ten anonymized and digitized tissue slides, which were stained with the H&E technique. After initial coarse-grained annotation, 625 non-overlapping tiles were extracted from contiguous tissue areas for each class. Each tile has the same size of 150×150 pixels (equivalent to 74 μm × 74 μm). The eight tissue classes are: tumor epithelium, simple stroma (homogeneous composition, includes tumor stroma, extra-tumoral stroma and smooth muscle), complex stroma (containing single tumor cells and/or few immune cells), immune cells (including immune-cell conglomerates and sub-mucosal lymphoid follicles), debris (including necrosis,

haemorrhage and mucus), normal mucosal glands, adipose tissue, background (no tissue). Here, for the sake of brevity, these classes are labeled as: *tumor*, *stroma*, *complex*, *lympho*, *debris*, *mucosa*, *adipose* and *empty*.

3.4. Methods

3.4.1. Model architecture

Using the building blocks described in **3.2**, a new neural network architecture was designed in conjunction with Marcin Możejko (also described in his MSc thesis). The architecture is called ARA-CNN.

The ARA-CNN network accepts RGB images of size (128, 128, 3) as its input (**Figure 3.6a**), where the values represent respectively: vertical resolution, horizontal resolution and the number of color channels. The images from the training dataset were downsized to these dimensions. Input values are propagated to the first part of the network called stem (**Figure 3.6b**). The stem contains a convolutional layer consisting of 64 filters, with filter size of (7, 7) and stride size of (4, 4). This is directly followed by max pooling with window size of (2, 2), and identically sized strides. The output from this part is of size (16, 16, 64), where the values are: reduced width, reduced height and the number of filters. These operations decrease the spatial dimensions by a factor of 8, which in turn significantly reduces memory usage and can be considered an adaptation of network topology to a relatively simple texture structure of the input [142].

The stem is followed by the first block (**Figure 3.6c**). The main aim of this part is to learn and extract initial discriminative low-level image features. It consists of 4 residual sections, where the input to each block is transformed by a convolutional layer with 64 filters - each sized (3, 3) and with stride of size (1, 1). The result of this convolution is added to its unchanged input, which creates a residual connection. The final section of this block is followed by an average pooling with window size of (2, 2). This makes the output of this part of the network to be shaped (8, 8, 64). The next part of the model is the second block (**Figure 3.6d**), which learns and extracts the final discriminative features - they are more high-level and abstract than the features in the first block. The structure of the second block is the same as that of the first block. After the final average pooling with window size (2, 2), the output from this part is of size (4, 4, 64).

The model has two outputs in total: the auxiliary output (**Figure 3.6e**) and the main

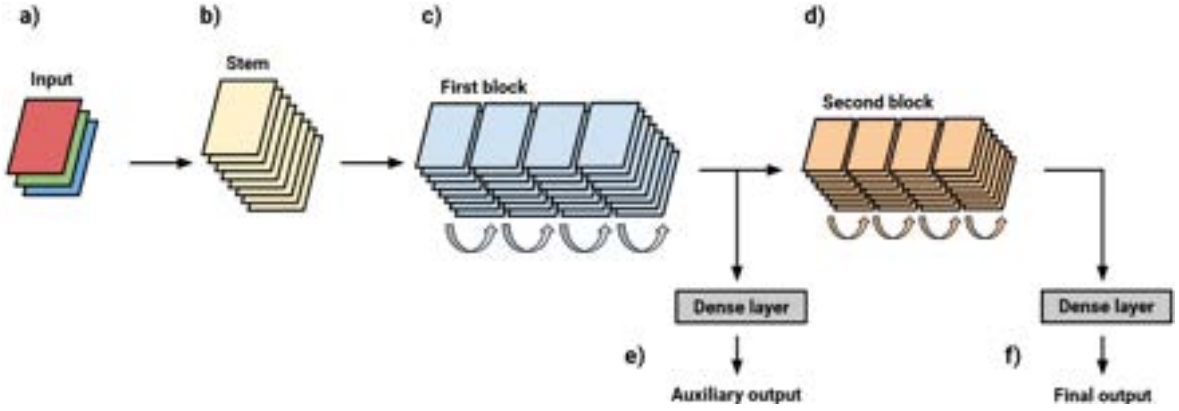

Figure 3.6: Structure of the ARA-CNN model. The network takes as input RGB images with dimensions of 128x128 pixels. They are passed to the stem, which contains a convolutional layer responsible for reducing the spatial dimensions of the input. This is followed by the first block and the second block, responsible for learning low-level and high-level image features respectively. Both of these blocks consist of four residual sections, with each of these sections containing a convolutional layer and a residual connection. The model has two outputs overall - an auxiliary output from the first block and a final output from the second block. Both of them use the softmax activation function.

output (**Figure 3.6f**). The main purposes of the former are to provide a better training signal to the stem and the first block (by making their features more discriminative) and to deal with the vanishing gradient problem [15] during training. The output from the first block is transformed by global average pooling. Next, it is transformed by a fully-connected layer with 32 filters and dropout with rate of 0.5. This dropout rate is usually the most effective in practice [20]. Finally, it is fed to a fully-connected output layer with a softmax activation function. The final output is used for making the actual predictions. After the second block, the data is transformed by exactly the same set of transformations as in the auxiliary output - global average pooling, then a fully-connected layer with dropout, followed by a final output layer.

Each layer in the network (except the outputs) connects to a Batch Normalization layer. The activation function used throughout the model is Leaky ReLU [159]:

$$f(x) = \begin{cases} x & \text{for } x > 0 \\ \alpha x & \text{otherwise,} \end{cases} \tag{3.15}$$

where the parameter α is set to 0.1 and x is a weighted sum of inputs to a network unit.

When it comes to the loss function, the categorical cross-entropy function was applied to both (auxiliary and final) outputs. The final loss was then expressed as a weighted sum of

these two losses with weight 0.9 for the final output and 0.1 for the auxiliary output.

3.4.2. Active learning with ARA-CNN

Active learning is an iterative procedure, where the initial model is trained on a small dataset and in consecutive iterations it is re-trained on a dataset extended by new samples. At each step, the new samples are added according to some acquisition function evaluated using the current model. Intuitively, the uncertainty measures described above are a good basis for an acquisition function in deep learning. In a given iteration, the model should first choose the samples it is most uncertain of [35].

Here, we propose an active learning framework, where the ARA-CNN model is used in conjunction with a pathologist in a feedback loop (**Figure 3.7a**). In a given training iteration, tissue patches are extracted from annotated regions and are used as a dataset for the model. During model evaluation, the uncertainty of each prediction is measured and most uncertain samples from the test dataset are added to the training data. At the same time, the pathologist is informed which classes are the most uncertain and is asked to annotate more regions for them. Additionally, we also detect individual test patches that were misclassified with a high certainty and pass them to the pathologist to verify. The whole process is repeated until a satisfactory level of performance metrics for the ARA-CNN model is reached.

We implemented and compared effectively three different acquisition functions. Two were based on uncertainty measures H and $BALD$, whereas the third was a random selection and served as a baseline. We performed a series of experiments in order to determine if uncertainty-based active learning can speed-up training with the colorectal cancer dataset. To this end, we emulated the proposed active learning workflow utilizing the available data. We started from generating three random splits of the full dataset - this gave us three test sets of 504 images and three training sets of 4496 images. Then for each of these test-train pairs, we performed the active learning procedure for both uncertainty measures plus a baseline training process based on random selection of images. In each case, we started from selecting 40 images per class (so 320 in total) from the training dataset. We trained the model on that small dataset and then, based on a given acquisition function, we chose 160 images to add to the previous 320. This slightly larger set became a new training dataset. We repeated this process, adding 160 images in each step, until there were no more images to draw from the initial full training dataset, giving us 28 training steps in total. Additionally, in order to

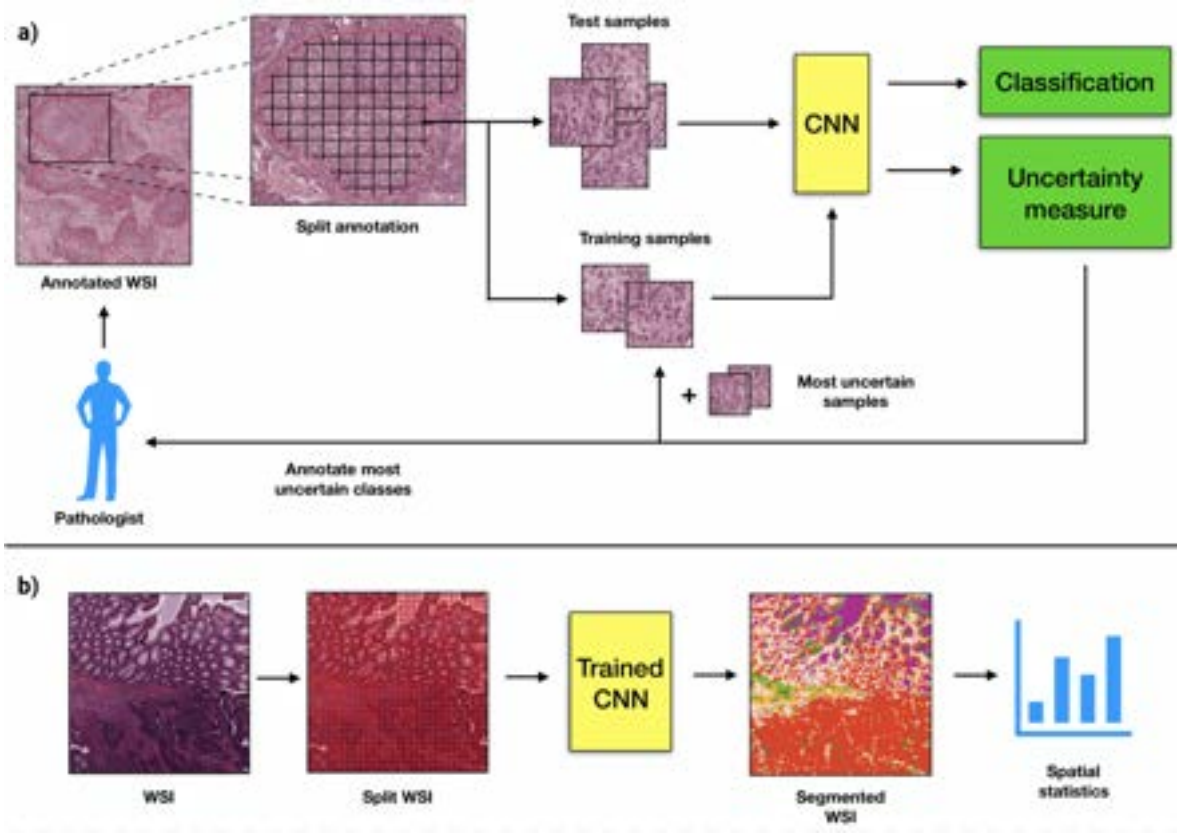

Figure 3.7: Overview of the proposed ARA framework. **(a)** Active histopathology workflow. Annotated whole-slide images (WSIs) are split into small image patches, which constitute a dataset. ARA-CNN is trained on that dataset. After the first round of training, the pathologist should be informed about i) which classes are the most uncertain and ii) which image patches are misclassified and highly certain, and thus identified as potentially mislabeled. The former should inform the pathologist about which classes to prioritize in the next round of annotation. The latter should inform about which image patches should be re-annotated with correct labels. New annotated whole-slide images are then taken and the workflow is continued until a satisfying level of classification accuracy is reached. **(b)** Segmentation workflow. Whole slide images are split into small image patches. Each of these is classified by the trained ARA-CNN model and is assigned a color based on its classification result. These colored tiles are merged together to form a segmented whole slide image and can be analyzed in terms of their spatial relationships. Each resulting tile has a measured uncertainty value as well, so pathologists can make an informed decision whether to take the automated segmentation as-is or to inspect it manually.

eliminate the effects of random weight initialization, we pre-initialized the model 8 times for each step and used these initializations for each of the 3 dataset splits. Thus, for each of the 28 steps we had to train the model 24 times.

For the random selection, the 160 new images in each step were sampled uniformly at random from the full training dataset. For the uncertainty-based functions, we performed inference (i.e. evaluation) on remaining images from the full training set in each step. We evaluated the uncertainty for each image using the H and $BALD$ measures, according to Equations 3.13 and 3.14. We sorted the results by uncertainty in descending order and selected the top 160. The results for each active learning step were averaged between initializations and dataset splits.

3.4.3. Image segmentation with ARA-CNN

To perform segmentation of test tissue slides from the Kather *et al.* [65] dataset, each of these 5000×5000 px images was split into 10000 non-overlapping test samples with resolution of 50×50 pixels (**Figure 3.7b**). These test images were then supplied as input to our model (by being upscaled to 128×128 pixels), which returned a classification into one of eight classes of colorectal tissue. Since the output of the model is a probability distribution, we selected the class with the highest value as the prediction for a given test image patch. We did not consider the measured uncertainty in this process. To get the final segmentation, we assigned a color to each predicted class and generated a 50×50 pixels single-colored patch for each test image. These patches were then stitched together to form the final images. Lastly, we applied a blurring Gaussian filter to smooth out the edges of tissue regions.

Finally, we performed a simple spatial analysis for each slide by counting the percentage of surface area taken by each class.

3.5. Results

3.5.1. Image features in H&E images

CNNs learn important image features from the training data. These features have a form of convolutional filters and can be easily visualized. In fact, such filters are analogous to activations observed in the mammalian primary visual cortex. This region of the brain has a spatial arrangement - it has a two-dimensional structure corresponding to that of the retina. As such, if for example only the lower half of the retina receives light, then only half of the primary visual cortex is activated [142]. To measure these activations, an electrode needs to be attached to a neuron in the visual cortex. By showing the subject several white noise images

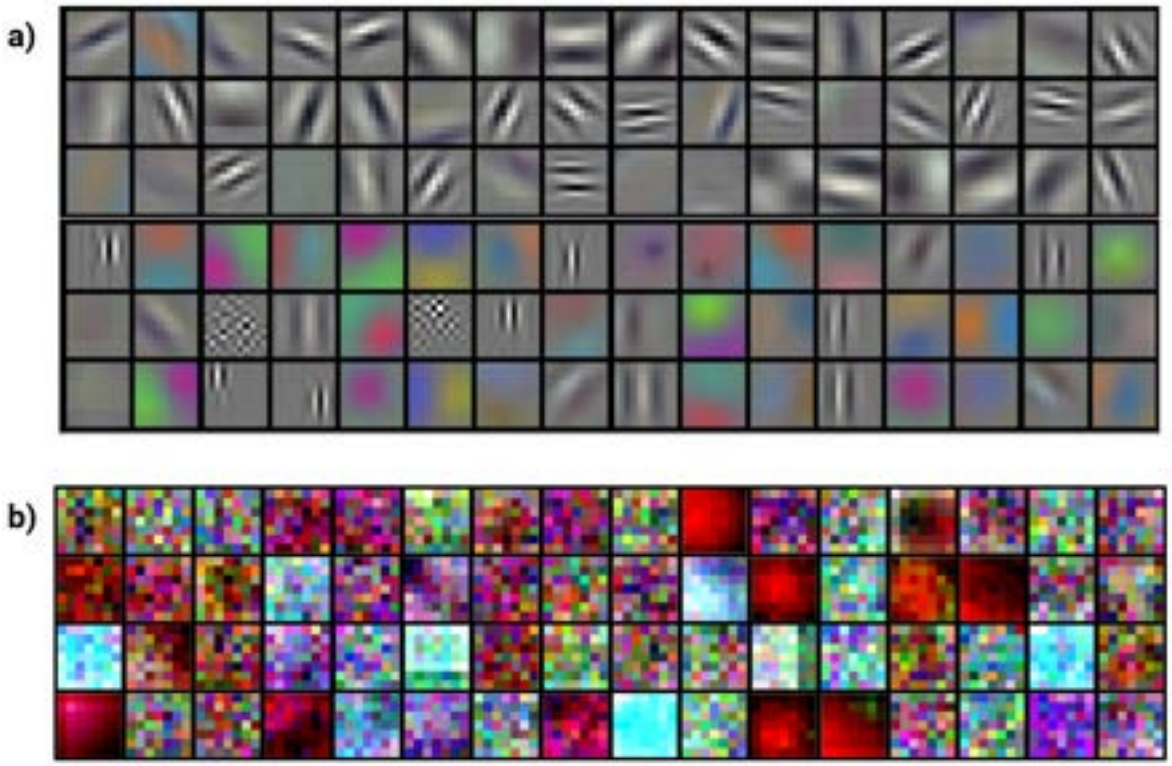

Figure 3.8: Comparison of convolutional filters generated for natural images and histopathological tissue patches. **(a)** Convolutional filters from AlexNet trained on the ImageNet dataset [11]. The features clearly represent edges, textures or colors. **(b)** Convolutional filters from ARA-CNN trained on the Kather *et al.* dataset of colorectal cancer. The features represent some patterns, but it is not obvious what they mean and why they are important.

and then fitting a linear model to the response signal, it is possible to approximate what could be understood as the neuron's weights in a process known as reverse correlation [160]. For CNNs, the assessment of filters is much easier, as the weights are learned by the model and easily accessible. By doing so, one can observe what features are most relevant for a given dataset. An interesting observation can be made if filters for natural images and histopathological tissue patches are compared (**Figure 3.8**).

CNNs trained on natural images learn filters that are easily identifiable. They represent various types of edges, patterns and colors (**Figure 3.8a**). More formally, they can be described by Gabor functions, similarly to weights extracted from neurons in the primary visual cortex [161]. Such a link is understandable, as CNNs trained on natural images represent what the mammalian visual system is evolutionally designed to do. With these filters, one can clearly see if e.g. trees are detected by vertical edge filters.

On the other hand, networks trained on histopathological images learn very complex filters (**Figure 3.8b**). It is not obvious what they represent, as the patterns are very chaotic. This means that the structure of H&E slides is much more complex than that of natural images, which implicates that the task of classifying tissue types is much harder than that of natural image classification. The difference can also be understood in terms of neurophysiology - because CNNs are based on the structure of the visual cortex, training them with images that do not normally occur in nature produces unfamiliar and complicated filters.

3.5.2. Model performance

To evaluate the performance of ARA-CNN, similarly to previous models trained on the same dataset, we measured its receiver operating characteristic (ROC) curves, area under the ROC curves (AUC) and error rates in 10-fold cross-validation for both 8-class and 2-class (*tumor* vs *stroma*) classification tasks. In addition, we also evaluated precision-recall curves. We used images with all color information preserved. The results were compared to those of the original model by Kather *et al.* (**Figure 3.9**), as well as to other methods that used the same dataset. Where necessary, we performed 5-fold or 2-fold cross-validation and used the results as a comparison point. In their work, Kather *et al.* [65] tested the performance of several low-level image features in combination with four classification algorithms, applied to grayscale images from their dataset. Their approach is an example of a 'traditional' procedure, where image features have to be hand-crafted and chosen appropriately depending on the dataset. The best results were reported for a combination of features containing: pixel value histograms, local binary patterns, gray-level co-occurence matrix and perception-like features. The best performing classifier was a support vector machine (SVM) algorithm with the radial basis function (RBF) kernel.

The ROC curves (generated with a one-vs-all method) for the 8-class experiment show excellent performance of ARA-CNN (**Figure 3.9a**). The AUC values for the *tumor*, *mucosa*, *lympho*, *adipose* and *empty* classes range from 0.997 to 0.999. Values for the *stroma*, *complex* and *debris* classes are a little lower (from 0.988 to 0.992), which indicates that the model cannot always distinguish them from other classes. Still, the mean AUC value is 0.995, which is higher than the value of 0.976 obtained by Kather *et al.* [65]. The ROC curve for the 2-class problem (**Figure 3.9b**) and its corresponding AUC value of 0.998 also illustrate near-perfect performance of ARA-CNN. It is important to note that performance evaluation using ROC

curves for the multiclass classification task in a one-vs-all setting may be biased due to the fact that the classes are unbalanced. In such a setting, it is better to use precision-recall curves (**Figure 3.9c**). The AUC values for these curves, as obtained by ARA-CNN, are a bit lower than for the ROC curves, but with the mean AUC of 0.972 are still indicative of excellent performance. The lowest AUC value (0.924) is obtained for the *complex* versus all classification task. This indicates that the *complex* class is the most difficult one to classify correctly for the model. We did not compare these results to other methods, as we were not aware of any other approaches that used precision-recall curves for performance evaluation on this dataset.

In terms of error rates, for the 8-class problem the ARA-CNN model reached an average rate of 7.56%, which is substantially lower, by 5.04 percentage points (p.p.), than the best result reported by Kather *et al.* [65] (**Figure 3.9d**). Similarly, in the binary classification task, we obtained an error rate of 0.89%, lower than 1.4% obtained by Kather *et al.* Thus, ARA-CNN is better than the best of standard approaches presented by Kather *et al.* [65], especially in the multiclass classification scenario. One of the differences between deep learning and the standard approaches is that the former construct the features on the fly based on the data itself. Here, the features identified by ARA-CNN as part of the learning process outperform the set of features that were engineered by Kather *et al.* [65] in the difficult task of decisively describing all classes in a multiclass image classification problem.

The classification performance of ARA-CNN is also superior or comparable to other published models that used the Kather *et al.* dataset, including both traditional and deep learning approaches that utilize CNNs (**Table 3.1**). ARA-CNN outperforms the traditional methods by a significant margin both in terms of AUC and accuracy. Ribeiro *et al.* [162] developed a traditional method that uses multidimensional fractal techniques, curvelet transforms and Haralick descriptors. They tested its performance using the Kather *et al.* dataset in a binary classification scenario, in which they reached an accuracy of 97.68%. ARA-CNN outperforms this result by 1.43 p.p.. When it comes to CNN methods, Wang *et al.* [163] performed 5-fold cross-validation and reported a mean AUC value of 0.985 (lower by 0.01 than ARA-CNN) and 92.6% accuracy (higher by 0.36 p.p. than ARA-CNN) for their BCNN in the multiclass task. Although BCNN and ARA-CNN achieve similarly high performance results, their architectures are very different. BCNN depends on an external method to perform stain decomposition of H&E images and is composed of two simple feed-forward CNNs, which take as input separate signals from the eosin and hematoxylin components and whose outputs are

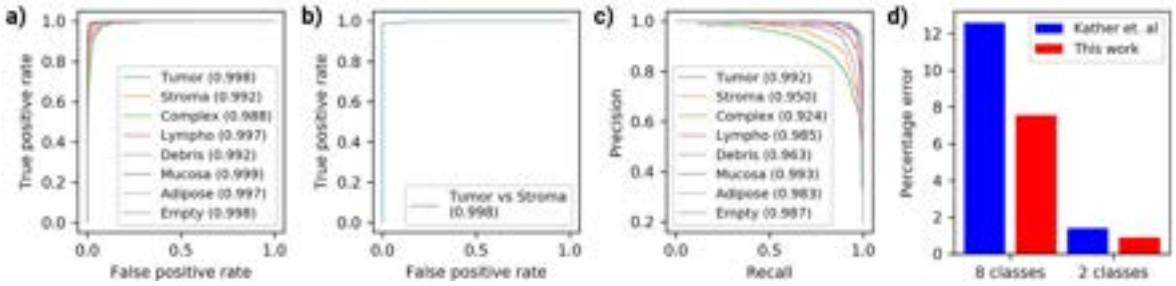

Figure 3.9: Model performance in 10-fold cross-validation. **(a)** ROC and area under the ROC curve (AUC) for classification into eight tissue types. The model presented in this work achieved an average AUC of 0.995 (a mean was taken across all eight classes). **(b)** ROC and AUC for binary classification between *tumor* and *stroma*. ARA-CNN achieves AUC of 0.998. **(c)** Precision-recall curves for ARA-CNN in a multiclass classification setting. The mean AUC for these curves is 0.972. **(d)** Error comparison to previous work. With error rate 7.56% for eight class classification, ARA-CNN substantially reduces the error (by 5.04 p.p.) compared to the error rate of 12.6% for the best model assessed by Kather *et al.* [65]. For binary (*tumor* versus *stroma*) classification, ARA-CNN has error rate 0.89%, which is also lower than the 1.4% error rate of the Kather *et al.* model.

combined by bilinear pooling. We took a more typical deep-learning approach, with a deeper network with residual connections, where no independent feature extraction nor decomposition is needed, and the network itself is responsible for extracting important signals from raw image data. Pham [164] used an autoencoder architecture to re-sample the images from the Kather *et al.* dataset and trained a small supervised network for different re-sampling factors. They reported at best an accuracy of 84.00% for binary classification, which is lower by 14.88 p.p. in comparison to our result. Ciompi *et al.* [165] used the Kather *et al.* dataset for testing their model trained on an independent colorectal cancer dataset and reported relatively small accuracies of 50.9% and 75.55%, where the former was achieved without stain normalization and the latter was an improvement resulting from having stain normalization applied. However, since this model was trained on a different dataset, we cannot directly compare our result to theirs. Overall, ARA-CNN achieves excellent performance on the Kather *et al.* dataset, and scores better than most other published methods that utilized the same data for training. Exceptional performance of our approach indicates that it successfully combines the flexibility typical for deep neural networks with strong regularization resulting from dropout and Batch Normalization.

Table 3.1: Comparison of different methods that used the Kather *et al.* dataset for training. ACC–accuracy. Trad.–traditional. Performance measures of compared methods as reported by the authors are summarized. Results in **bold** are the best in their category.

* The authors do not explicitly state the number of folds. Since in other reported results the number of folds they used is 10, we assume 10-fold cross-validation here as well.

Method	**Type**	**Problem**	**Max. reported 10-fold ACC**	**Max. reported 5-fold ACC**	**Max. reported 2-fold ACC**	**10-fold AUC**	**5-fold AUC**
Kather et al.	Trad.	Binary	98.6%	-	-	-	-
		Multi	87.4%	-	-	0.976	-
*Ribeiro et al.**	Trad.	Binary	97.68%	-	-	-	-
Sarkar et al.	Trad.	Multi	73.66%	-	-	-	-
Wang et al.	CNN	Multi	-	**92.6 ± 1.2%**	-	-	0.985
Pham	CNN	Binary	-	-	84.00%	-	-
ARA-CNN	CNN	Binary	**99.11 ± 0.97%**	**98.88 ± 0.52%**	**98.88%**	**0.998**	**0.999**
		Multi	**92.44 ± 0.81%**	92.24 ± 0.82%	**88.92 ± 1.95%**	**0.995**	**0.995**

3.5.3. H&E slide segmentation

In histological image analysis, the labeling of image patches is only the first step in the process of segmentation. To get the full overview of a tissue slide, it is necessary to see how image patches of different classes are placed in relation to each other and to measure their relative abundance. In particular, it is interesting to determine the neighborhood of tumor cells. For example, the tumor being infiltrated by immune cells may be a marker of good prognosis. Kather *et al.* [65] showed a simple segmentation approach using standard classification methods. We present a recreation of their procedure using the ARA-CNN model (see **3.4.3**).

We performed segmentation of five full tissue slides from the Kather *et al.* [65] dataset (**Figure 3.10**). The segmentation can obviously be improved - the approach with stitching

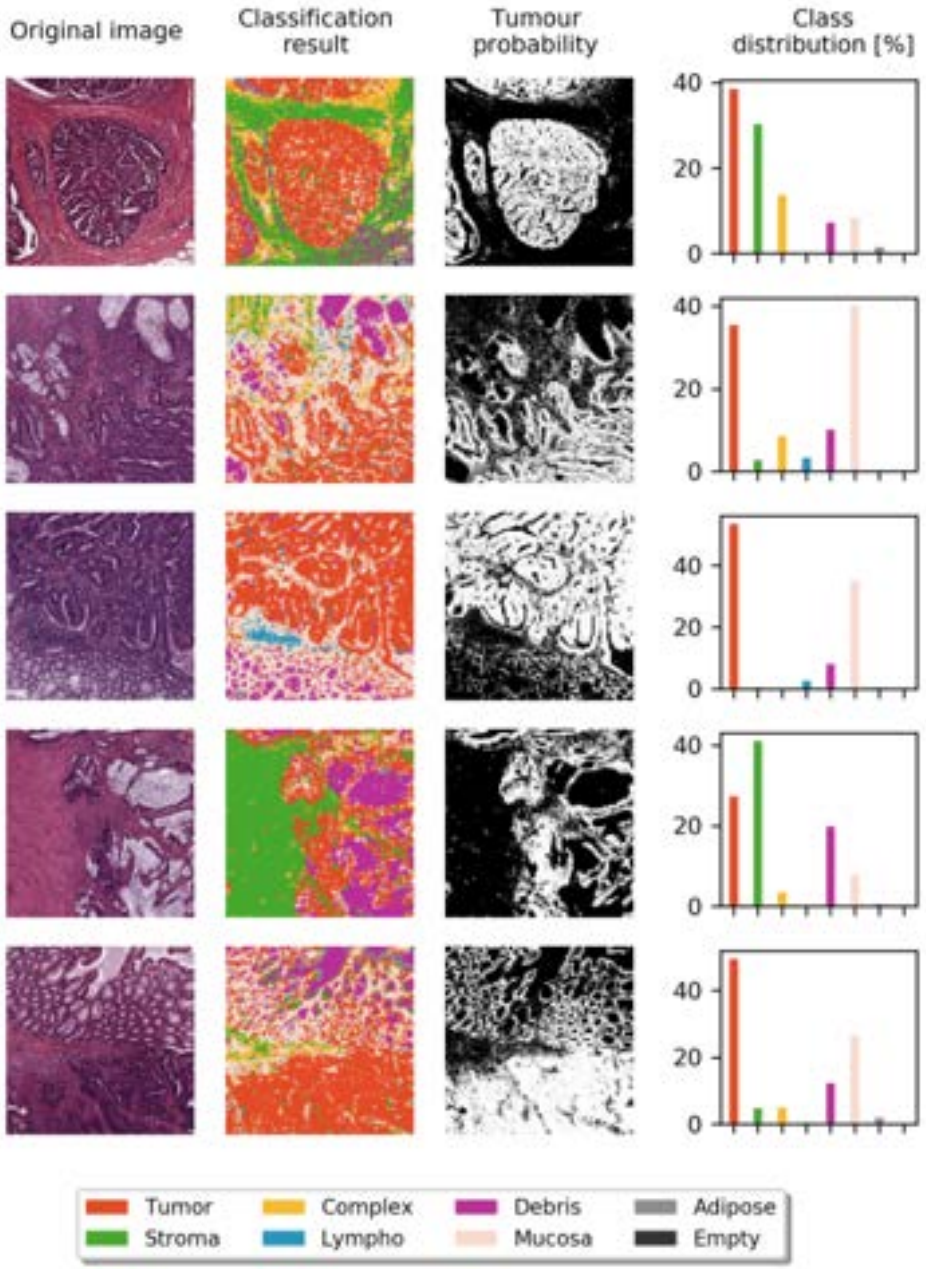

Figure 3.10: Segmentation of five large tissue slides from the colorectal cancer dataset, performed with ARA-CNN. The leftmost column presents the original H&E slides, the second one shows the segmentation done with ARA-CNN, while the third one is a visualization of the *tumor* class probability (the lighter the segment, the more probable that there is a tumor there). The last column contains a class distribution histogram - each bar represents the percentage of a given class in the segmented image.

image patches is after all quite rudimentary. However, it can be good enough to see the aforementioned spatial relationships. As a basic spatial statistic, for each slide we generated a summary of tissue class distribution (**Figure 3.10**). Histograms such as these can be used as a filter to find images for further consideration (e.g. those with high tumor concentration) in an automated diagnosis system.

3.5.4. Uncertainty, active learning and identification of mislabeled images

Uncertainty analysis

The model presented in this work, thanks to its implementation of dropout and variational inference, has a few ways to measure the uncertainty of each prediction. These uncertainty measures allow the model predictions to be reliable. Consider an example image, which is classified by the model as *tumor* with high probability 0.95, but the measured uncertainty is also high. This can mean that the prediction cannot be taken for granted and needs to be double-checked by a human. Here, we evaluated two uncertainty measures, Entropy H and $BALD$, checking their distribution in each class and their performance as acquisition functions in active learning on the Kather *et al.* dataset.

First, we applied the trained model to 504 test images. For each image, we recorded the classification and the measured uncertainty. For Entropy H, on average, the highest uncertainty values were reported for images from the *stroma* and *complex* classes (**Figure 3.11a**). The biggest variance in uncertainty was measured for the *debris* class. These three classes were also misclassified as each other, which indicates that they are similar in appearance and the model has a hard time differentiating them. This is in agreement with the previously described precision-recall curves (**Figure 3.9c**) and with the analysis described below. In addition, it can be observed that misclassification occurred almost exclusively when the uncertainty was high. Thus, a high uncertainty is indeed a good indicator that the prediction may be faulty. For $BALD$, on average, the most uncertain classes according to that measure are *stroma* and *complex*, in agreement with Entropy H (**Figure 3.11b**). Interestingly, $BALD$ measured much less variance in the *debris* class, which makes *lympho* the most variable class in this case. Moreover, the *empty* class is relatively more certain according to $BALD$ than in the Entropy H experiment. These differences may be a result of epistemic and aleatory uncertainties present in the data, which are measured differently by $BALD$ and Entropy H. Nevertheless, the $BALD$ measure still captures the fact that misclassifications take place

mainly for highly uncertain predictions.

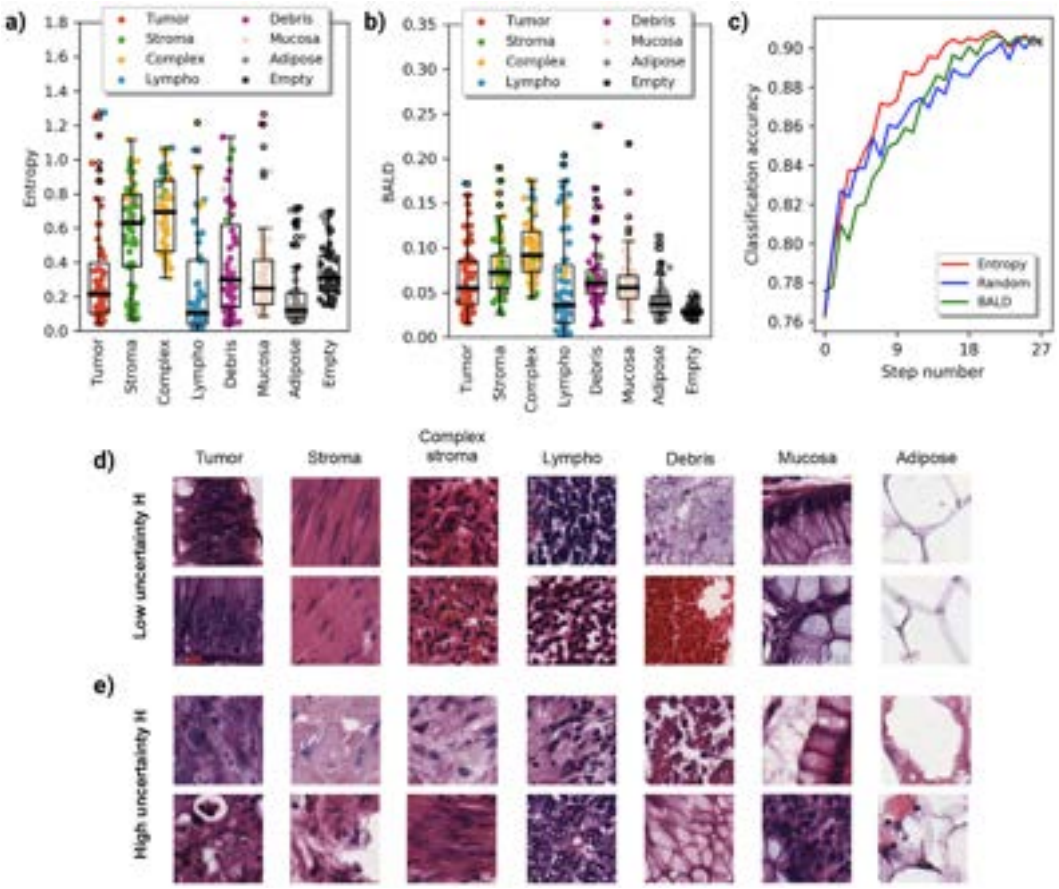

Figure 3.11: The uncertainty of image classification. **(a, b)** Distribution of uncertainty for the colorectal cancer images used to train the ARA-CNN model. The horizontal axis shows the actual class of these images, whereas the classification of each image is represented with colored jitter. The y-axis value represents the amount of uncertainty. ARA-CNN is on average most uncertain when it comes to the *stroma* and *complex* classes. It also makes mistakes in classification mostly when it is uncertain. **(a)** Distribution of uncertainty for the Entropy H measure. **(b)** Distribution of uncertainty for the $BALD$ measure. **(c)** Results of active learning experiments. Starting from a small training dataset with 320 images in total (step number 0, 40 images per class), the model was re-trained on the dataset enlarged in every iteration by 160 additional images. Three distinct acquisition functions were tested: Random, Entropy H and $BALD$. At each step, the average classification accuracy was measured (y-axis). **(d, e)** Microscopic images of tissues composing colorectal cancer and its microenvironment. Samples were categorized by uncertainty measured with Entropy H. Columns correspond to different tissue classes. **(d)** Images with low uncertainty H. **(e)** Images with high uncertainty H.

Active learning

We designed an active learning process (see **3.4.2**) with either the Entropy H or $BALD$ measures acting as acquisition functions. We evaluated its efficiency on the Kather *et al.* dataset by analyzing the resulting model accuracy as a function of the number of training samples (**Figure 3.11c**). The Random acquisition function served as a baseline. In initial active learning iterations the Entropy H measure performed very similarly to random selection, but from step 7 (which contained 1440 images) Entropy H achieved consistently higher accuracy (with on average 2% improvement in classification accuracy) until the very end of the process. The accuracy of the model trained on samples selected using the $BALD$ measure was worse than the random one from the start of active learning until step 12. From step 13 (which contained 2400 images) it got slightly better, but never eclipsed the accuracy received using the Entropy H measure. This proves that the Entropy H uncertainty measure can be successfully used as an acquisition function in active learning scenarios utilizing the ARA-CNN model. It can speed up the learning process by roughly 45%. The model reached the classification accuracy equivalent to the full dataset already at step 15, in which the training set contained 2720 images. Thus, the fraction of images required for obtaining the full accuracy is only 2720 out of 5000 (54.4%), and the fraction of steps required is only 15 out of 27 (55.56%), both amounting to around 45% reduction. It means that this subset of images, chosen based on the Entropy H uncertainty measure, was large enough to accurately train the model.

Identification of mislabeled images

We propose that images that are misclassified by ARA-CNN with high certainty (i.e., $H < H_t$, where Ht is a predefined threshold value) are good candidates for identifying mislabeled training samples. To demonstrate the performance of our identification approach, we artificially introduced increasing percentage of mislabeled images into the training set and measured sensitivity and specificity, while recording the overall model performance.

To this end, we randomly divided the dataset into a training set and a test set, with the same proportions as during the model training. Next, we took the training set, randomly sampled a given percentage p_m of images and changed their assigned class at random. Finally, we trained the model on a training dataset with these mislabeled images reintroduced. We defined the set of positives P as candidate mislabeled images identified by our approach. The set of true positives TP was defined as all of the artificially mislabeled images. Sensitivity was

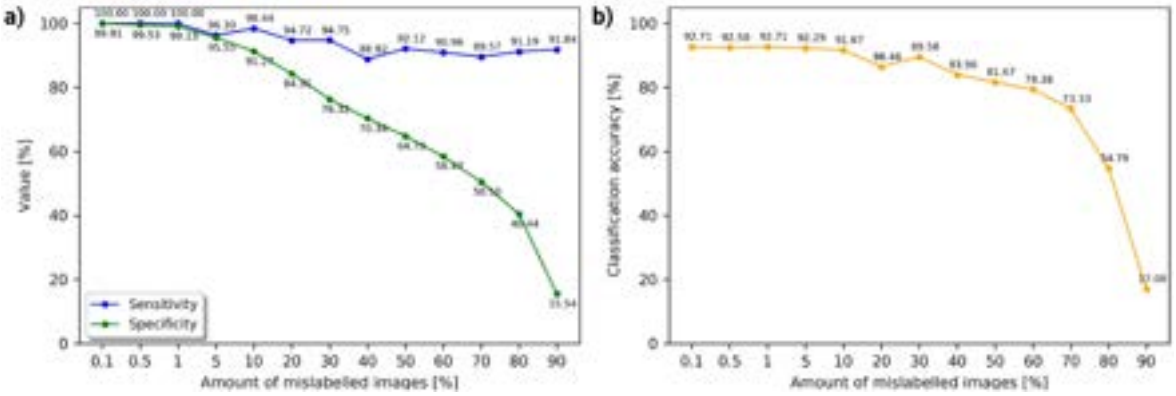

Figure 3.12: Identification of mislabeled images as a function of their percentage in the training set. **(a)** Sensitivity and specificity for the proposed mislabeled sample identification strategy. **(b)** Classification accuracy of ARA-CNN decreases only for very high fraction of mislabeled training images.

evaluated as $|TP|/|P|$ and specificity as $|TN|/|N|$, with TN and N being the complement sets for TP and P, respectively.

Sensitivity of mislabeled image identification is overall very high, and is only slightly affected by the growing percentage of mislabeled samples (**Figure 3.12a**). For $p_m \in \{0.1, 0.5, 1\}$, sensitivity is at 100%, meaning that all misclassified images with uncertainty below H_t are in fact mislabeled. For higher p_m values, sensitivity is slightly lower, but it never dips below 88.82% (for $p_m = 40$). Specificity decreases with the increase of the percentage of mislabeled training samples, but remains at a very high level even for substantial percentage p_m, dropping below 80% only at p_m around 20%. This demonstrates that uncertainty H can be used to find mislabeled training samples even when the noise in the training data is extremely high.

We also measured what effect an increasing p_m has on the classification accuracy of ARA-CNN on the test set (**Figure 3.12b**). Remarkably, up to and including 5% of artificially mislabeled training samples, the performance is not affected. From 10% up to and including 70%, it decreases, but only slightly. From 80%, the amount of mislabeled images is too large and the model cannot be trained properly, which results in a substantial drop in accuracy. Such a good classification performance even in the case when the majority of the training samples are mislabeled at random indicates that ARA-CNN is highly robust to noise in the training data.

This result is characteristic of Bayesian neural networks in general. These models learn representations that are more sparse than non-Bayesian methods, which explains their ability to generalize in noisy label scenarios. Regularization provided by dropout causes the neurons

to be less influenced by mislabeled samples. Fewer free parameters mean that label noise is not over-explained (i.e. the network has less capacity to overfit), so Bayesian neural networks create representations that are less prone of overfitting to mislabeled samples [166]. In addition, the utilization of variational inference, which is in essence an ensemble procedure, causes lower model volatility (defined as a standard deviation of a neuron's activations over a dataset) and lower responsiveness of the network's neurons. This contributes to the robustness of the model, as opposed to using dropout only during training [166].

Understanding the uncertainty of image classification

To investigate what pathological features of images are determinant for assigning specific uncertainty values measured by Entropy H, we selected test images with very low ($H \leq 0.2$) and very high ($H \geq 0.8$) uncertainty and asked a trained pathologist to inspect them by eye. We focused on Entropy H due to its superior performance in active learning. There were no examples of the *empty* class with high uncertainty, indicating this class is easy for the algorithm to recognize and classify properly. For each of the remaining seven tissue classes, images of lowest uncertainty display characteristic pathological features (**Figure 3.11d**). Images of the *tumor* class with low H display cells that have distinct changes in their nuclei: enlargement, hiperchromasia (dark violet color), improper chromatin distribution (i.e. spots with higher and lower density) accompanied by multiplication of nucleoli, increased nuclear to cytoplasmic ratio, nuclear fusions and cell overlapping. The images of the *stroma* class with lowest uncertainty display typical uniformly stained pink, eosinophilic fibers with elongated nuclei, and low nuclear to cytoplasmic ratio. For the images of the *complex* class with low assigned H, the stroma is infiltrated by lymphatic or neoplastic cells with addition of erythrocytes. The highly certain images of the *lympho* class show features typical for areas of lymphocytic dense infiltration - lymphocytes are intensively stained, monomorphic cells with round nucleus and very scarce thin, basophilic cytoplasm rim. Nucleoli are not visible. Images of the *debris* class with low uncertainty H values are composed of various tissue samples. First, they contain a mucous, amorphic substance creating multiple, fine vesicles, white in the center with violet contours. On top of that, features characteristic of the *debris* class are mostly extravasated erythrocytes – red, round cell conglomerates presenting very dense collocation with blurred cell contours. Images of the *mucosa* class with very low assigned uncertainty show typical features of mucosal glands in the large intestine. They are composed

of visible characteristic goblet cells that are cylindrical in shape and contain big, round areas filled with mucus - white with violet margin. Small, regular, dark nuclei are visible at the cell periphery. Goblet cells lay in linear or rosette–like formations. Finally, images of the *adipose* class with low uncertainty show pathological features typical of the adipose tissue. They are composed of big, white polygonal areas with violet, wide contours, adhering to each other tightly. No nuclei are visible.

In contrast to low uncertainty images, the images with the highest uncertainty show features that are pathologically difficult to categorize (**Figure 3.11e**). For very uncertain images of the *tumor* class, the sparse cells visible within the stroma show fewer features of malignancy – most of them are small, regular in shape, with no visible nucleoli. No nuclear fusions or overlapping cells are observed. The pictures could be mistaken with complex stroma. For the images of the *stroma* class that were assigned very high uncertainty H, the tissue has irregular structure without typical linear fibers and elongated nuclei. Empty spaces in both example images and very low color intensity in the top one may be artifacts, although whole samples could be categorized as complex stroma or perchance debris because of listed alternations. Out of the two complex stroma example images with very high uncertainty, in the top image (**Figure 3.11e**, third column) there are no visible fibers. At the same time, the image contains many pale vesicular areas slightly similar to mucus. The bottom image could be interpreted as a normal stroma sample, because of its color, fibrotic structure and shape of the nuclei. In the top image representative of very high uncertainty images of the *lympho* class, cell arrangement is not very dense and there is a lot of stroma visible between nuclei – this could be categorized as complex stroma instead. The bottom picture shows many features of malignancy that should suggest diagnosis of tumor cells. From the two uncertain example images from the *debris* class, the top consists of tissue residues with no particular structures visible. The bottom image shows structures very similar to mucosal glands – areas of mucus are bigger and well margined in comparison to amorphic mucus specific for this category. From the two high uncertainty images of the *mucosa* class, the top image has heterogeneous composition. In the right part of the image, goblet cells with their nuclei can be seen. The left part is full of amorphic substance and could be categorized as debris. In the bottom example, only the lower left corner looks like mucosal glands forming a rosette. The rest of the image contains stroma with lymphatic infiltration, thus pathologically could be categorized as complex stroma. In the top uncertain example of the *adipose* class, although white, empty spaces are clearly visible and cell walls have more irregular margins than normally. In the

bottom example, the characteristic polygonal shapes are not visible. The images do not suit any other category more than adipose tissue, however they do not share its typical important features.

The above interpretative analysis of tissue images was performed in collaboration with Dr Joanna Zambonelli.

3.6. Conclusions

We implemented an accurate, reliable and active machine learning framework for histopathological image classification. Its most crucial part was a new Bayesian deep learning model, ARA-CNN. The model was applied to the task of colorectal tissue classification and incorporated it into an uncertainty-based active pathology workflow. The classification accuracy achieved by our model exceeds the results reported by authors of the training dataset by Kather *et al.* [65] used in this work. The proposed CNN architecture showed outstanding performance in both binary and multiclass classification scenarios, reaching almost perfect accuracy (error rate of 0.89%) in the former case and best in class (error rate of 7.56%) in the latter. It also surpassed the classification performance of other methods that were trained with the same dataset by up to 18.78 p.p..

Thanks to its Bayesian nature, ARA-CNN is capable of measuring uncertainty of its predictions. This allowed us to implement an active learning procedure in which using the Entropy H uncertainty measure as an acquisition function yielded a 45% reduction in training time required to achieve maximum possible accuracy. Additionally, we showed that the H measure produced by ARA-CNN can be used as a basis for detecting label noise in training data. We proved that our proposed mislabeled sample identification strategy is effective even for very noisy datasets.

The excellent performance of ARA-CNN indicates that it is a step forward in establishing accurate and reliable machine learning models for histopathology. Furthermore, due to its deep learning nature, the ARA-CNN architecture easily handles datasets that contain tissue types other than the colorectal tissue used in this project. These characteristics of ARA-CNN were exploited in the project described in the next chapter.

Chapter 4

Deep learning-based spatial features predict patient survival and gene mutations in lung cancer

4.1. Lung cancer fundamentals

4.1.1. Genetic mechanisms

In principle, cancer is a genetic disease [167]. Due to many possible factors (exposure to toxins, radiation, lifestyle, familial predisposition, among others) DNA in human cells undergoes mutations at an increased rate. If DNA repair mechanisms are not able to fix all of them, they are again propagated in the process of transcription to RNA molecules, which then pass this erroneous information to proteins during translation [168]. This incorrect genome expression causes improper cell function, which characterizes all tumor cells. If these cells are not eliminated by the immune system or killed in the process of apoptosis (in which corrupted cells are signaled to die), they start to multiply, which causes tumor growth (**Figure 4.1**). Moreover, if the initial alteration occurred in an oncogene (which normally promotes cell growth) or in a tumor suppressor gene, then this multiplication can get out of control. These processes can happen in any part of the body and ultimately characterize the cancer type. That is, e.g. if tumor cells appear in lungs, the patient can develop lung cancer.

Identifying and studying such carcinogenic DNA mutations has been a backbone of cancer research for a long time. Over the last decade, many tumor types have been sequenced from

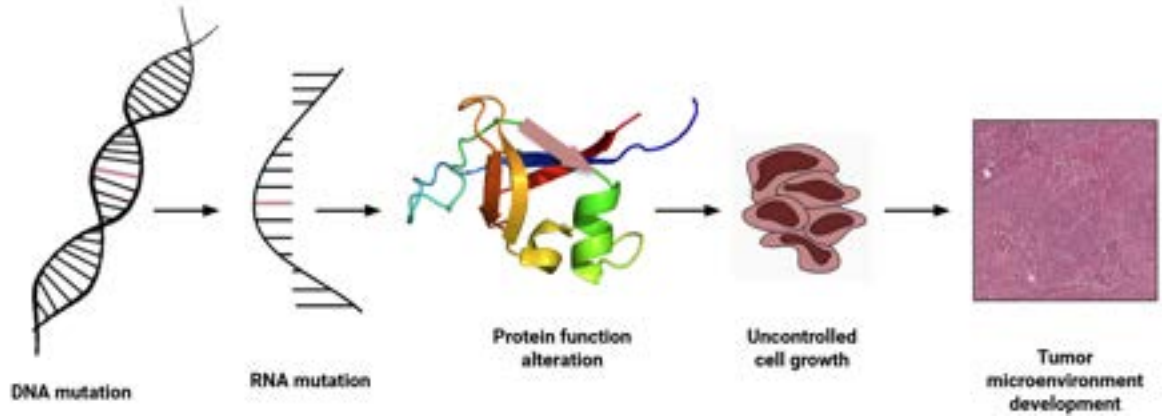

Figure 4.1: Cancer progression. DNA mutations cause alterations in protein function, which can lead to development of tumor cells and their growth into tumors.

patients, which showed that cancer is a very heterogeneous disease [169, 170, 171, 172, 173]. Unfortunately, there are many thousands of such possible mutations, which is a huge issue for cancer research. This amount of alterations makes any kind of statistical modeling extremely complex, and as such it is hard to connect a specific mutation with patient phenotype. In this work, we narrowed this issue and studied relationships between pre-selected alterations and the phenotype in the form of histologic tumor microenvironment.

4.1.2. Lung cancer

According to the latest WHO statistics [89], lung cancer is the second most diagnosed and leading in terms of deaths cancer type worldwide. Worryingly, due to increased tobacco smoking in developing countries, the number of new cases is growing on a yearly basis. As such, lung cancer study is of critical importance to public health.

There are two main types of lung cancer: non-small cell lung cancer (NSCLC) and small cell lung cancer (SCLC). The former is much more prevalent, accounting for 85% of all cases [174]. NSCLC can be further categorized into adenocarcinoma (LUAD), squamous cell carcinoma (LUSC) and large cell. Adenocarcinoma is the most common of the three, accounting for 40% of all lung cancers [175]. Squamous cell carcinoma constitutes 25% to 35% of lung cancers, while large cell cancers accounts for 5% to 10% of lung cancers [174].

Next to routine chest x-ray scans, the main diagnostic method used for lung cancer is the histopathological analysis of H&E tissue slides [174]. Such slides are routinely stored for each patient, so there is an abundance of patient-specific, disease progression-relevant data that

until recently has not been utilized at scale in cancer research. This has changed with the advent of digital pathology and the development of machine learning approaches to various predictive tasks based on digitized H&E image data [50, 52].

In this context, a particularly critical avenue of research is to study the relationship between the organization of the tumor microenvironment and both patient survival and gene mutations in lung cancer, which is now feasible more than ever thanks to the advancement of deep learning methods.

4.2. Analyzed data

4.2.1. Clinical samples

We obtained the formalin-fixed paraffin embedded (FFPE) tissue samples from 55 primary tumors of lung cancer (35 LUAD, 20 LUSC). The material was derived from FFPE surgical resections at the Medical University of Lublin, Poland. The tasks of acquiring the samples and their experimental analysis were performed by Dr Marcin Nicoś and Dr Tomasz Kucharczyk from the Medical University of Lublin, Poland.

At the moment of diagnosis and surgical resection of the primary cancer lesions, none of the patients had received neoadjuvant therapies. Clinical and demographic patients' data was collected in a manner that protected their personal information. The study protocol received ethical approval from the Ethics Committee of the Medical University of Lublin, Poland (no. KE-0254/235/2016).

4.2.2. Hematoxylin and eosin staining

FFPE tissue was cut into 3 µm fragments and placed on glass slides. The slides were then stained with hematoxylin and eosin in a Leica Autostainer XL device (Leica Biosystems, USA), with pre-staining xylene and ethanol wash steps, 7 minutes of hematoxylin, and 20 seconds of eosin. Post-staining wash steps were also included in the program. The glass slides were cover-slipped directly after the staining procedure.

4.2.3. Glass slides digitalization

The H&E stained slides were scanned using the Aperio ScanScope CS2 device (Leica Biosystems, USA) equipped with a 20x Olympus microscope lens. The images were stored on an

internal server in a form of SVS files, which were further analyzed.

4.2.4. Training dataset for ARA-CNN

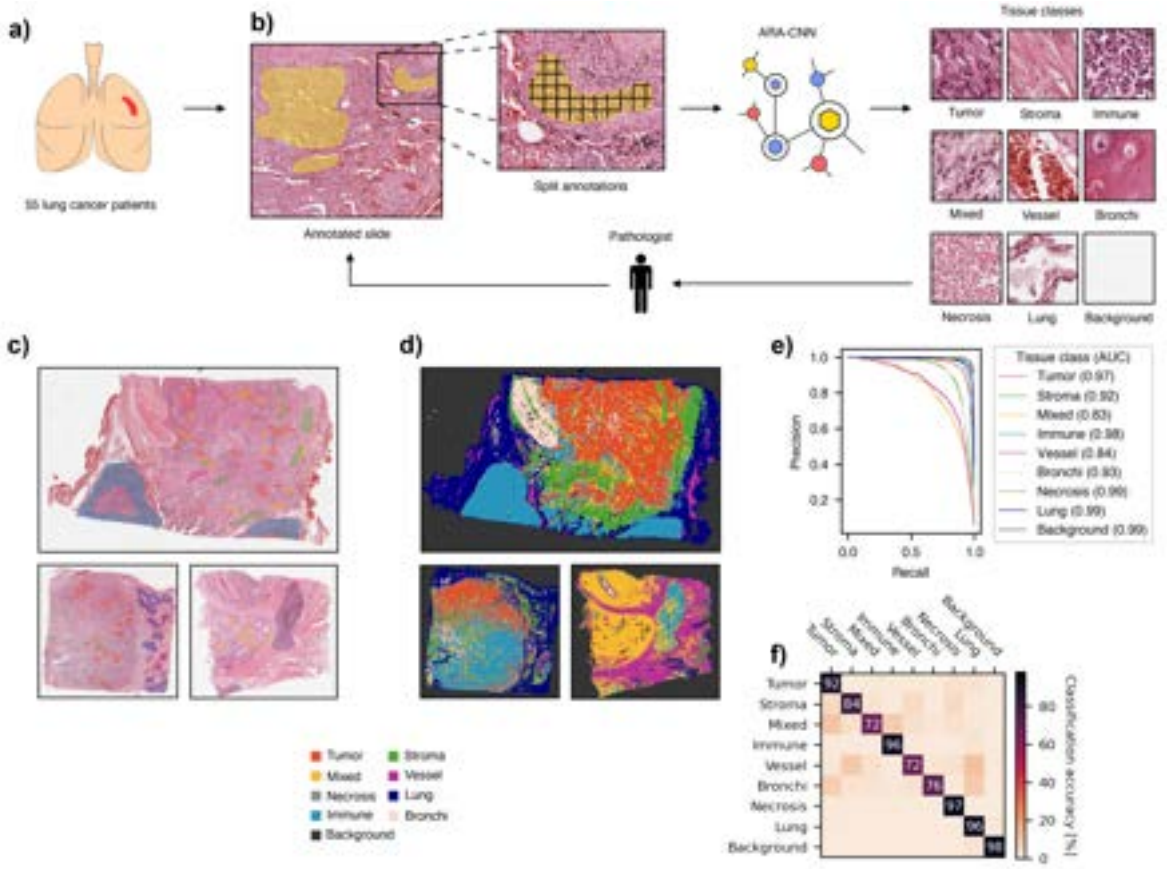

Figure 4.2: Overview of training ARA-CNN for lung cancer tissue classification. **(a)** We sourced H&E tissue slides from 55 lung cancer patients. **(b)** These slides were annotated by an expert pathologist in an active learning loop with ARA-CNN, which resulted in the *LubLung* dataset and a trained tissue classification model. **(c)** Example annotations of various tissue regions. **(d)** Segmentation results from ARA-CNN show that tissue heterogeneity in the TME is captured correctly. **(e)** Precision-recall curves for each tissue class obtained in a 10-fold cross-validation scheme on the *LubLung* dataset. The mean AUC was 0.94. **(f)** Confusion matrix for ARA-CNN trained with *LubLung*. Row labels indicate true classes, while column labels describe classes predicted by the model.

We extracted the training dataset from the H&E slides sourced from 55 lung cancer patients, with 1 slide per patient (**Figure 4.2a**). Regions of contiguous tissue were annotated using QuPath [176] by an expert pathologist, Dr Iwona Paśnik from the Medical University of Lublin, who marked them as one of the following nine classes: *tumor* with neoplastic epithelial cells; *stroma* composed of connective tissue within tumor or extra-tumoral connective

tissue; *mixed* where connective tissue was strongly infiltrated with immune cells; *immune* composed of lymphocytes and plasma cells or fragments of pulmonary lymph nodes; *vessel* composed of smooth muscle layers (veins and arteries) with red blood cells within lumen; *bronchi* composed of cartilage and bronchial mucosa; *necrosis* including necrotic tissue or necrotic debris; *lung* (lung parenchyma); and *background* of the tissue scan (no tissue). The TME in the original slides differed between patients, which gave us a diverse set of training examples (**Figure 4.3**). Some of the slides were more covered by tumor and necrotic cells or stroma, while in others immune infiltration, vessels or mixed class were dominating. In most of the slides we observed the "normal" lung structures, so bronchi was less common and needed more training data from many sections. All annotated regions were chosen for the purpose of providing the best material for model training. To this end, for a given class, the pathologist was annotating tissue that was undoubtedly of that class, and there was enough of that tissue visible in the slides to provide enough annotated patches. For example, for the vessel class, we did not consider arterioles, dilated capillaries or venules, as these tissues were too small for the chosen patch scale. Lymphatics were ignored due to them being imperceptible on H&E slides. For the immune class, intrapulmonary lymph nodes were included due to their high concentration of well-visible lymphocytes, even though the presence of such lymph nodes is not correlated with the tumor's immune response.

The annotated regions were then traced by a moving window, which cut out non-overlapping square patches of tissue with side size of 87 µm. In addition to 87 µm, we also tested the training performance for patches with sizes of 74 µm and 100 µm (see **4.4.2.** below). They resulted in worse performance, so we proceeded with using the 87 µm sized ones. This gave us an initial version of the training dataset, which was then improved upon by utilizing human-in-the-loop active learning, as part of the accurate, reliable and active (ARA) image classification framework, described in the previous Chapter [123] (**Figure 4.2b**). In total, we ended up with 23199 patches, divided in the following manner: 3311 *tumor* patches, as well as 1511 *stroma*, 716 *mixed*, 1196 *immune*, 1236 *vessel*, 2030 *bronchi*, 4448 *necrosis*, *6031* lung, and 2211 *background* patches.

4.2.5. TCGA data extraction and processing

Independent patient data, including H&E images, mutations, and clinical information was extracted from the TCGA database (up to date as of 2020-05-14) through a REST API

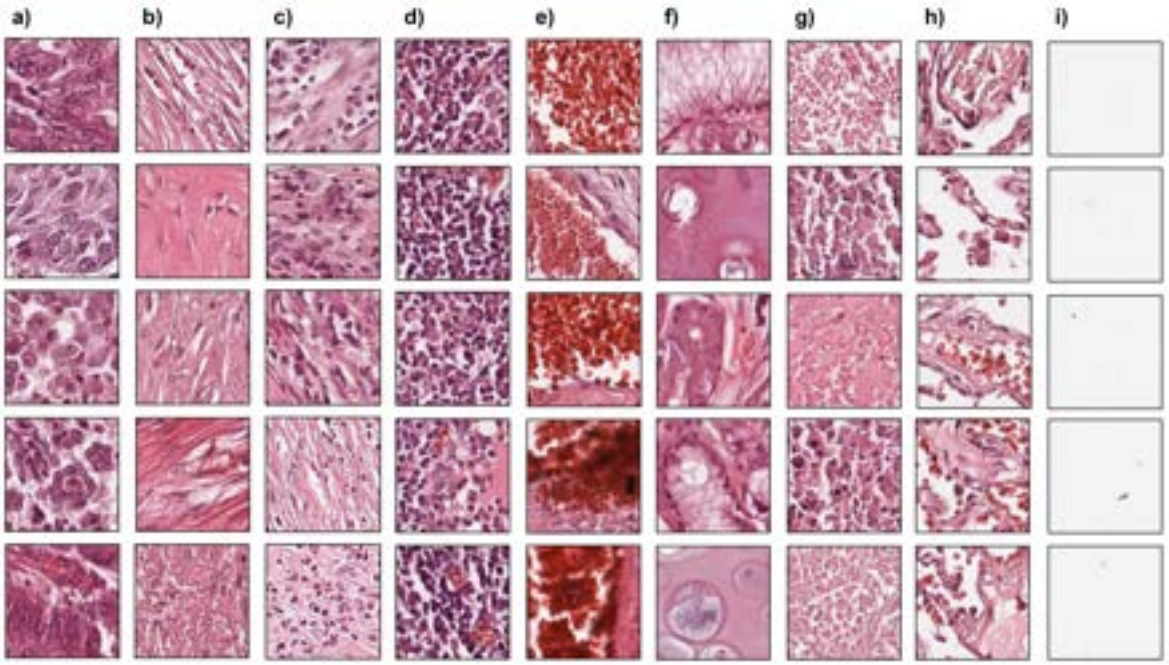

Figure 4.3: Example image patches from *LubLung*. **(a)** The *tumor* class. **(b)** The *stroma* class. **(c)** The *mixed* class. **(d)** The *immune* class. **(e)** The *vessel* class. **(f)** The *bronchi* class. **(g)** The *necrosis* class. **(h)** The *lung* class. **(i)** The *background* class.

provided by TCGA. The database contained 478 LUAD cancer patients with at least one H&E tissue slide per patient. Out of these, frozen tissue slides were filtered out, which left 514 images. We developed a parallelized pipeline that downloaded the slides, ran all necessary calculations and then removed the processed images.

Each processed slide was split into non-overlapping patches with side size of 87 µm, same as for the patches in *LubLung*. As an optimization step, we filtered the extracted patches and excluded the ones where most of the area was empty. To perform this filtering, we first converted each patch to grayscale using the standard Rec. 601 luma formula:

$$I(x,y)_Y = 0.299 \cdot I(x,y)_R + 0.587 \cdot I(x,y)_G + 0.114 \cdot I(x,y)_B \tag{4.1}$$

where $I(x,y)_Y$ is the grayscale converted pixel value at position (x,y), and $I(x,y)_R$, $I(x,y)_G$, $I(x,y)_R$ are the red, green and blue components of that pixel, respectively. Then, we mapped each pixel to either black ($I(x,y)_Y < 200$) or white ($I(x,y)_Y \geq 200$) and counted them. The patch was deemed as relevant if the proportion of black pixels to white pixels was larger than 0.05.

In addition to the slides, clinical and mutation data was also extracted from TCGA. The former was downloaded with the curatedTCGAData R package [177] and it contained data

for 518 LUAD patients. However, we removed data for Asian patients, as they are noted to be very distinct when it comes to disease progression [178, 179]. This left us with clinical data for 510 patients. Mutation data for 563 LUAD patients was downloaded from the UCSC Xena Browser [180] (dated 07-20-2019), in the form of a *TCGA-LUAD.muse_snv.tsv* file. In addition, it was downsized to include only genes selected as relevant to lung cancer. To this end, we examined a set of genes that are either known to be frequently mutated in lung cancer and are important for patient prognosis and treatment, as characterized by the SureSelect Cancer All-In-One Catalog and Custom Assays [181], or were studied previously by Kather *et al.* [111]. Specifically, we selected such genes from the Kather *et al.* study, which showed an AUC higher than 70% or p-value < 0.001 in the task of mutation prediction by a deep learning model. From this set of 24 genes, we filtered out such genes for which there were only up to 15 LUAD patients carrying their mutation. This resulted in the following set of 13 selected genes: *ALK*, *BRAF*, *DDR2*, *EGFR*, *KEAP1*, *KRAS*, *MET*, *PIK3CA*, *RET*, *ROS1*, *STK11*, *TP53*, *PDGFRB*.

All datasets were merged together by the TCGA patient identifier. This left us with an intersection between the image, clinical and mutation datasets, which contained 444 patients and 506 slides.

As a last step, the clinical variables were pre-processed in the following manner. Age was quantified into two groups: 65 years and older, as well as younger than 65 years. Sex was set to 1 for male and 0 for female patients. Pack years were quantified into three groups: non-smoker (0 pack years or smoking history set as 'lifelong non-smoker' or, if information about pack years was missing, smoking history set to 'current reformed smoker for $>$ 15 years'), light smoker (less than 30 pack years or, if information about pack years was missing, smoking history set as 'current reformed smoker for $<$ or $=$ 15 years') and heavy smoker (30 or more pack years, or smoking history set to 'current smoker'). Pathologic stage was mapped into three groups as well: early (stage I, Ia, Ib), locally advanced (stage II, IIa, IIb, IIIa) and advanced (stage IIIb, IV).

4.3. Methods

4.3.1. Training and validation of the ARA-CNN model

For this new dataset, we used exactly the same architecture as the one described in Chapter 3, with the only difference being the number of output classes (nine here instead of eight).

Thanks to the fact that we were able to work closely with a pathologist, we were able to train the model in an active learning procedure.

The model was trained in several iterations, each one improving upon the previous ones. After the first training process, the distribution of uncertainty for images in each class was measured separately and due to higher median uncertainty, it was concluded that there are three classes in need of more training examples: *mixed*, *vessel* and *bronchi*. These were passed on to the pathologist, who labeled new regions belonging to these classes. The resulting new training samples were extracted and added to the previous training dataset. This adaptive training procedure was repeated three times, where each time the uncertainty for each class was measured, until it was decided that the uncertainty results were at a satisfactory level.

For training and evaluation of the ARA-CNN model on the lung cancer tissue patches, we used stratified 10-fold cross-validation. In each fold, the whole dataset of 23199 images was split into a training dataset and a test dataset used for evaluation. The test dataset contained 2316 patches, while the training dataset consisted of 20883 patches. Each class was split in exactly the same proportion: 10% were sent to the test dataset and 90% to the training dataset.

Additionally, in each training epoch the training data was split into two datasets: the actual training data and a validation dataset. The latter was used for informing the learning rate reducer - we monitored the accuracy on the validation set and if it stopped improving, the learning rate was reduced by a factor of 0.1. This split was in proportion 90% to 10% between actual training data and the validation set, respectively.

For parameter optimization, we used the Adam optimizer [153]. The training time was set to 100 epochs. The training data was passed to the network in batches of 32, while the validation and test data was split into batches of 128 images. The loss function used during training was the categorical cross-entropy. The final model was trained on the whole dataset of 23199 patches.

4.3.2. TCGA H&E patch normalization

Due to the fact that the tissue slides stored in TCGA exhibit high color variation, they needed to be normalized to a common color space, matching that of the training dataset. Three normalisation algorithms were considered: Reinhard *et al.* [182], Macenko *et al.* [183] and Vahadane *et al.* [184]. To decide which of these three should be used on the TCGA data,

a series of experiments was conducted, in which the *LubLung* training dataset was normalized with each of these algorithms and then ARA-CNN was trained in a cross-validation scheme. The results showed that the best classification performance (mean accuracy 81.52% for Macenko *et al.*, 82.39% for Vahadane *et al.*, 85.76% for Reinhard *et al.*) was achieved for the dataset variant normalized with the Reinhardt *et al.* algorithm. Consequently, each relevant patch extracted from the TCGA slides was normalized individually with the Reinhardt *et al.* procedure. The normalization was performed with a region of interest image selected at random from the training dataset. All image patches from the TCGA database were transformed to match the color space of that image.

4.3.3. TCGA image data segmentation using ARA-CNN

The normalized patches served as input to ARA-CNN. For each input patch, the model returned a classification probability into each of the nine predefined classes. With these results, each patch was labeled as the class with the highest probability and then the labeled patches were merged back into their full respective slides and colored by the label. This created segmented slides, with clearly visible continuous areas of differing tissue.

The segmented slides were next validated by an expert pathologist, Dr Iwona Paśnik, who assessed that 39 slides needed to be excluded from further analysis. There were two reasons for that. The first one involved erroneous classifications returned by ARA-CNN - 21 out of 506 slides contained errors of such nature. The other 18 slides were excluded due to markings and other staining errors. After this process, the final dataset contained 467 slides from 411 patients.

4.3.4. Quantification of spatial features for the segmented tumor tissues

The obtained segmented images from TCGA were then processed further in order to extract spatial information in the form of two types of features, which we refer to as *tissue prevalence* (TIP) and *tumor microenvironment composition* (TMEC). TIP is a distribution of tissue classes within the whole tissue area, i.e. excluding the background class. TMEC measures a distribution of tissues that neighbor the tumor tissue within a predefined margin.

The prevalence t_i of tissue i is expressed as:

$$t_i = \frac{n_i}{N}, \tag{4.2}$$

where n_i is the number of patches for tissue i, N is the total number of tissue patches (excluding the *background* class) and $i \in \{tumor, stroma, mixed, immune, vessel, bronchi, necrosis, lung\}$. The vector T with entries given by t_i makes up the TIP features. The *background* class was omitted, as it is not relevant to the tissue structure.

The microenvironment composition m_j for tissue j is:

$$m_j = \frac{b_j}{B}, \tag{4.3}$$

where b_j is the number of patches of class j that neighbor the *tumor* class and B is the total number of all patches neighboring the *tumor* class (excluding the *tumor* itself and the *background* class), with $j \in \{stroma, mixed, immune, vessel, bronchi, necrosis, lung\}$. The TMEC features are organized in a vector M, with m_j as its entries. The neighbor patches are considered only within a margin around the borders of tumor regions. Each tumor patch is considered separately and up to eight neighbors around it are counted. These patches are summed up to b_j for each class j and to B in total.

Using the microenvironment and prevalence data, we also calculated three spatial metrics that were previously defined in the literature: intra-tumor lymphocyte ratio (ITLR) [39], Simpson diversity index [185], Shannon diversity index [186]. We used a simplified version of these metrics - instead of cell-wise, we calculated them patch-wise. Specifically, these metrics were computed as follows:

$$\begin{aligned} ITLR &= \frac{b_{IMMUNE}}{n_{TUMOR}} \\ Shannon &= -\sum_i t_i \log(t_i) \\ Simspon &= \sum_i t_i^2 \end{aligned} \tag{4.4}$$

where b_{IMMUNE} is the number if immune patches that neighbor the tumor and t_{TUMOR} is the number of tumor patches in the whole slide.

4.3.5. Multivariate survival modeling using the Cox model

The aforementioned predictors were used as input to the Cox proportional hazards model [187]. They were organized into the following basic variants: clinical, clinical + ITLR, clinical + Shannon diversity index, clinical + Simpson diversity index, clinical + TMEC, clinical + TIP, clinical + TMEC + TIP. In addition, variants with mutation data added on top of clinical data were considered. Each variant was trained 50 times in a 10-fold cross-validation scheme, from

which the median c-index [188] values were aggregated. For categorical variables, the hazard ratio of their basal values was set to 1. For the sex variable, the basal value was 'Female'. For the Stage variable, the basal value was 'Early stage'. For mutation variables (*EGFR*, *STK11* and *TP53*) the basal value was the absence of alteration. Finally, for smoking status, non-smoker was set as basal. The survival analysis was done in cooperation with Michał Kukiełka as part of his MSc thesis.

4.3.6. Mutation classification

The processed data from TCGA served as input in the mutation classification task. The predictor variables were the same as in the survival prediction task (minus mutation status). The response variables were binary and were defined by the mutation status for the 13 previously chosen frequently mutated LUAD genes.

For each classification task, where the class was specified by the presence of mutation of a given gene, the dataset was oversampled so that positive (mutation occurred) and negative (mutation did not occur) subsets of examples were equal in size. Oversampling was done by inserting multiple copies of the positive examples so that their number reached that of the negative ones.

Eight combinations of predictive features were tested: clinical, clinical + ITLR, clinical + Shannon diversity index, clinical + Simpson diversity index, clinical + TMEC, clinical + TIP, clinical + TMEC + TIP, TMEC only. To classify the mutation status for each gene, two distinct machine learning models were trained and compared. The first one was a simple linear model in the form of regularized logistic regression. It was fitted using the Liblinear solver [189], with the L2 penalty and up to 2000 iterations. The second one was the Random Forest algorithm [190]. We used the implementation from the sklearn Python library [191] with default parameter values.

All models were trained 100 times with 10-fold cross-validation and the resulting classification accuracy metrics were averaged. Classification accuracy was evaluated using the AUC metric.

4.3.7. CRF definition

To improve the quality of segmentations produced by ARA-CNN, we added the Conditional Random Field (CRF) [113] model as a post-processing step applied to segmented slides. CRF

is a type of an undirected probabilistic graphical model. It is expressed by an undirected graph $\mathcal{H}$, in which nodes represent $Y \cup X$, where Y is a set of target output variables and X is a disjoint set of observed variables [192]. The graph is parameterized by a set of factors. We define a factor as a function $\phi : Val(D) \to \mathbb{R}^+$, where D is a set of random variables that is known as the scope of the factor. For $\mathcal{H}$, factors are $\phi_1(D_1), \ldots, \phi_m(D_m)$, with scopes $D_i \nsubseteq X$ (meaning that variables in D_i cannot exclusively be a subset of X). Two variables in $\mathcal{H}$ are connected by an edge if they are together is the scope of some factor [192]. A conditional distribution $P(Y|X)$ is encoded by the nodes of $\mathcal{H}$ as such:

$$\begin{aligned} P(Y|X) &= \frac{1}{Z(X)}\tilde{P}(Y,X) \\ \tilde{P}(Y,X) &= \prod_{i=1}^{m}\phi_i(D_i) \\ Z(X) &= \sum_{Y}\tilde{P}(Y,X), \end{aligned} \tag{4.5}$$

where $Z(X)$ is a so-called partition function, which is used for normalization.

Factors $\phi(D)$ can be converted to log-space:

$$\phi(D) = \exp(\phi'(D)) \tag{4.6}$$

The $\phi'(D)$ functions are known as potential functions. Furthermore, they can be expressed in terms of feature functions f:

$$\phi'(D) = \exp\left\{\sum_{k} w_k f_k(D)\right\} \tag{4.7}$$

where $w_k \in \mathbb{R}$ are parameter vectors and $f_k : Val(D) \to \mathbb{R}$. This representation is particularly useful for distributions with large domains, such as text or images [192].

4.3.8. CRF formulation for tissue segmentation

Because CRF is a graph-based model, it can be configured to match the structure of the tissue segmentation problem. Let's consider an input 2D grid of image patches X and an output 2D grid of image patches Y, placed on the grid based on their localization in the tissue. Each input patch x_i in X belongs to one of nine previously defined classes (see **4.2.4**), where $i \in S$, with S being the set of grid positions (i.e. patch coordinates in a segmented slide). Similarly, ground truth labels for patches y_i in Y are also defined for these nine classes. The model is meant to accept some input grid and transform it into an output grid, based on the learned

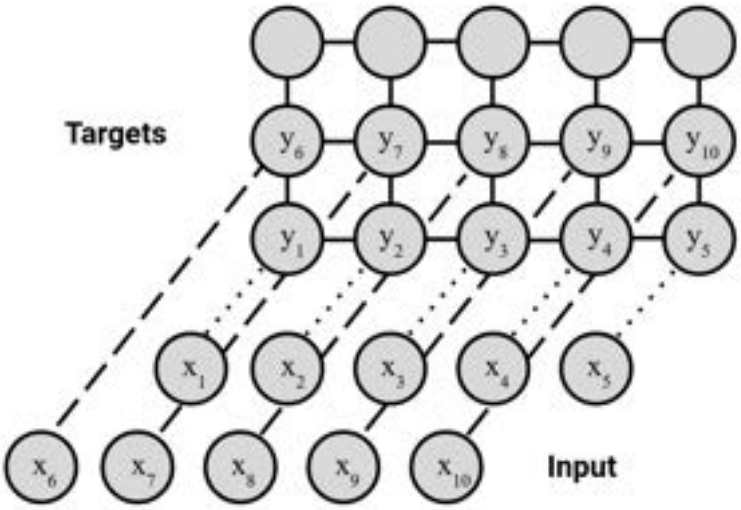

Figure 4.4: Structure of a 2D grid CRF model for tissue segmentation. The model learns a conditional distribution that maps from input X to targets Y. The neighborhood structure of the targets encoded by pairwise potentials ϕ'_{ij} informs the training process.

structured mapping $P(Y|X)$. With this, the CRF model for tissue segmentation can be expressed as such:

$$\begin{aligned} P(Y|X) &= \frac{1}{Z(X)}\tilde{P}(Y,X) \\ \tilde{P}(Y,X) &= \exp\left\{\sum_{i\in S}\phi'_i(y_i|X) + \alpha\sum_{i\in S}\sum_{j\in N_i}\phi'_{ij}(y_i,y_j|X)\right\} \\ Z(X) &= \sum_{Y}\tilde{P}(Y,X), \end{aligned} \tag{4.8}$$

where ϕ'_i are unary potentials, ϕ'_{ij} are pairwise potentials, N_i is a set of neighbors for patch i and α is a coefficient for defining the effect of pairwise interactions [193] (**Figure 4.4**). Because patches are placed on a 2D grid, we consider $|N_i| \in \{4, 8\}$. If we express ϕ'_i and ϕ'_{ij} in terms of feature functions, then $\tilde{P}(Y,X)$ can be reformulated as:

$$\tilde{P}(Y,X) = \exp\left\{\sum_{i\in S}\sum_{k,l=1}^{9} w_{ikl}f_{ikl}(y_i|X) + \alpha\sum_{i\in S}\sum_{j\in N_i}\sum_{k,l=1}^{9} w_{ijkl}f_{ijkl}(y_i,y_j|X)\right\} \tag{4.9}$$

The unary feature functions f_{ikl} are defined as such:

$$f_{ikl} = \begin{cases} 1 \text{ if } x_i = k \text{ and } y_i = l \\ 0 \text{ otherwise} \end{cases} \tag{4.10}$$

The pairwise feature functions f_{ijkl} are defined like so:

$$f_{ijkl} = \begin{cases} 1 \text{ if } y_i = k \text{ and } y_j = l \\ 0 \text{ otherwise} \end{cases} \tag{4.11}$$

Because the matrix for f_{ijkl} is symmetric, we consider only its lower triangular matrix.

In order to train the above model, we need to optimize some objective function with respect to parameters w. The process of training is described below.

4.3.9. CRF training

There are several approaches to training CRF models. The first one sets as objective the maximization of the conditional likelihood of the training data:

$$\max_w \sum_{i=1}^{N} \log(P(y_i|x_i, w)), \tag{4.12}$$

where N is the number of data points in the training dataset and w are model parameters.

Another popular approach is maximum margin learning [194]. In this case, the CRF is treated as a more general structured prediction problem. In such a setup, inference for a data point x takes the form of:

$$h(x, w) = \arg\max_{y \in Y} w^T f(x, y) \tag{4.13}$$

where y is a structured label, f is a feature function and w are model parameters in a convex parameter space. With this definition, the parameters w are learned by minimizing the loss-based soft-margin objective:

$$\min_w \frac{1}{2}||w||^2 + C \sum_{i=1}^{N} r(x_i, y_i, w) \tag{4.14}$$

where C is a regularization parameter and r is an upper bound on the empirical loss $\mathcal{L}$:

$$r(x_i, y_i, w) = \max_{y \in Y}(\mathcal{L}(y_i, y) + w^T f(x_i, y)) - w^T f(x_i, y_i) \tag{4.15}$$

The algorithm often used to solve the above optimization problem is subgradient descent. Here, we trained the CRF model in the maximum margin learning setting and used stochastic subgradient descent as an optimizer [194].

4.3.10. CRF experiment setup

To test the effectiveness of the above approach, a dataset needed to be be prepared that contained both input X and ground truth Y segmentations. Ideally, we wanted X to contain segmentation output from ARA-CNN and Y to be comprised of pathologist-annotated regions. To this end, we segmented 24 out of 57 slides contained in *LubLung* with ARA-CNN (trained with the full version of *LubLung*) in the same way as described before (see **4.3.3**). We selected these slides because they were the only ones with tissue regions annotated by a pathologist. We limited all further analysis exclusively to these tissue regions, as we did not have any other ground truth information for the slides in *LubLung*. As such, patch labels produced by ARA-CNN were taken as input X, while patch labels taken directly from expert annotations were treated as ground truth Y.

The model was trained in a 10-fold cross-validation scheme, where folds were split between slides. In other words, the training set and test set always contained different slides. Each of the segmented slides was divided into non-overlapping square slices of size 10x10 patches. To exclude from both training and evaluation all tissue regions outside annotated areas, the slices were filtered. If at least one patch inside a given slice was a part of an annotated region, that slice was selected for the actual dataset. All other slices were rejected. The final datasets comprised exclusively of such segmentation slices (**Figure 4.5**).

The CRF model was implemented with the *pystruct* [195] Python library. The α coefficient was set to 1 in all cases, so unary and pairwise feature functions were equally important. We set $|N_i| = 4$, so for each patch we considered 4 neighbors: left, right, top, bottom. The inference algorithm used in all experiments was AD3 [196]. The optimizer used to fit the model was Subgradient SSVM [194, 197], with the regularization parameter C set to 0.1, and the number of iterations set to 200.

4.4. Results

4.4.1. Study setup

We performed a multi-step study with the main goal of verifying whether it is possible to find a relationship between:

1. human-interpretable spatial composition features in H&E images, specified by the prevalence of different tissue types in the entire image and in the TME

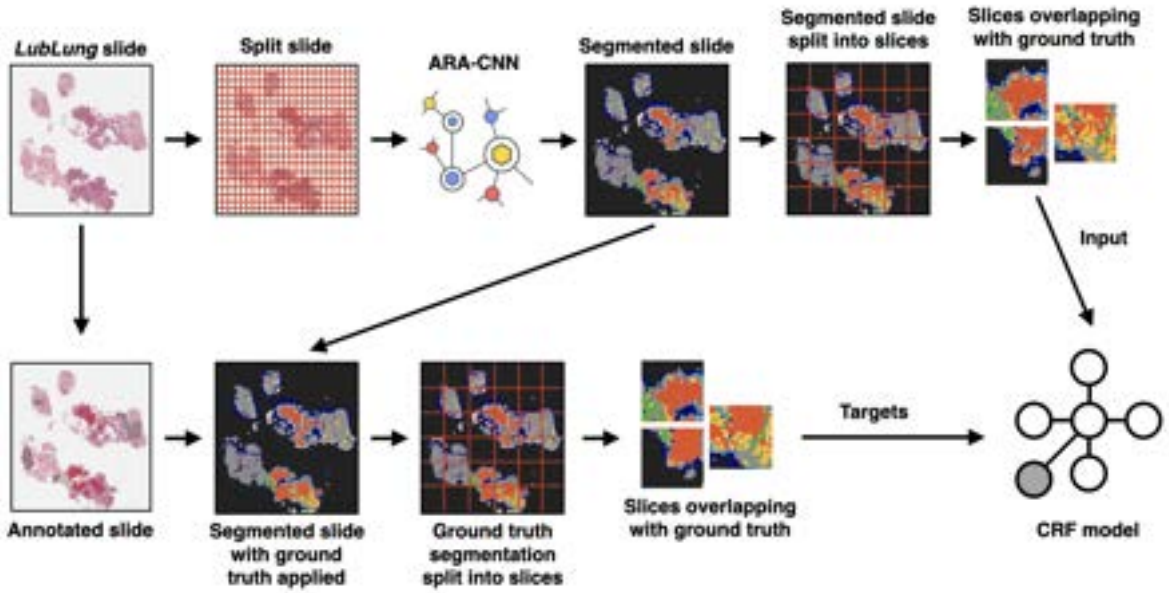

Figure 4.5: Training setup of the CRF model. Each of the 24 considered *LubLung* slides was segmented with ARA-CNN. These segmented slides were then corrected using existing tissue region annotations. Both corrected (ground truth) and original segmentations were then split into slices with the size of 10x10 patches. Slices overlapping with annotated areas served as the final training dataset for the CRF model.

2. both patient survival and gene mutations in lung cancer

The first step was the preparation of a new training dataset for tissue classification in lung cancer, *LubLung*, and training of a novel tissue classifier for this cancer (**Figure 4.2**).

In the second step, we applied the final ARA-CNN model to classify patches from 467 lung cancer slides extracted from the TCGA database (**Figure 4.6a**) and then created slide segmentations, from which we calculated human-interpretable spatial features.

In the third step, we extracted clinical and mutational features of the patients and their tumors, respectively, and then utilized them together with the H&E image spatial composition features for the 411 considered TCGA patients in two machine learning prediction tasks (**Figure 4.6c**).

4.4.2. Validation of ARA-CNN

To quantify the classification performance of ARA-CNN, we performed a series of experiments. First, we inspected which patch size is the most appropriate for *LubLung* slides. We tested the training performance for patches with sizes of 74 µm, 87 µm and 100 µm. The

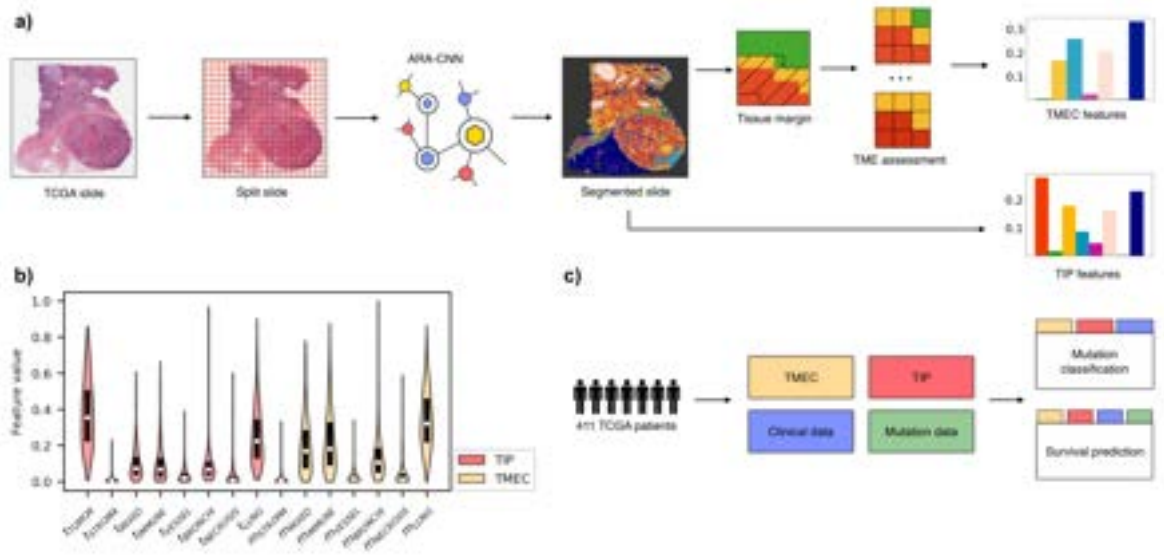

Figure 4.6: Calculation and utilization of TIP and TMEC features. **(a)** H&E slides from TCGA were downloaded and split into tissue patches. Each patch was classified with ARA-CNN, producing tissue segmentations. These segmentations were next used to calculate the TIP and TMEC features. **(b)** Distribution of individual component features in TIP and TMEC. The most often occurring features for TIP were t_{TUMOR} and t_{LUNG}. For TMEC, these were m_{LUNG}, m_{IMMUNE} and m_{MIXED}. **(c)** Tasks performed with the help of the TIP and TMEC features. In addition to the TIP and TMEC features, clinical and mutation data was also sourced from TCGA. These datasets were combined and served as input in two tasks: survival prediction and gene mutation classification. The results were compared to those obtained using previous spatial metrics instead of TIP and TMEC.

mean classification accuracy was 84.64% for 74 µm, 85.21% for 87 µm and 84.35% for 100 µm (**Tables 4.1, 4.2, 4.3**). Based on this result, we proceeded with the 87 µm patch size. Next, we verified how well the final trained model performs in segmenting the whole set of *LubLung* H&E slides. The segmentation allowed to correctly capture the TME heterogeneity in terms of all trained classes, which was confirmed by an expert pathologist, Dr Iwona Paśnik, who compared the original H&E slides with the final output of the model (**Figure 4.2c,d**). Next, we used a 10-fold cross-validation procedure on the final set of 23,199 annotated patches obtained in the *LubLung* dataset. The best performance in a single class versus rest classification was achieved for the *background*, *lung*, *necrosis*, *tumor*, and *immune* classes (area under the curve, AUC range: 0.97–0.99) (**Figure 4.2e**). The lowest AUC (0.83) was obtained for the *mixed* class, which is not surprising given that it is a tissue that is a mix of two other classes (*stroma* and *immune*). We then computed a confusion matrix, which confirmed that the best trained classes were *background*, *necrosis*, *lung*, *immune* and *tumor* (accuracy range:

92.36%–98.01%) (**Figure 4.2f**). In terms of errors, the model most often confused the *mixed* class with *tumor* (9.72% of the patches annotated as mixed were classified as tumor) or *immune* (8.17% of the patches); the *vessel* class with *stroma* or *lung* (8.73% and 10.79% of the patches, respectively); and the *bronchi* class with *tumor* or *lung* (7.30% and 8.53% of the patches, respectively). Given that patches of these classes were also often hard to distinguish by an expert pathologist, we conclude that our trained ARA-CNN model can reliably classify different tissue types in H&E images of LUAD and LUSC tissue sections.

Table 4.1: Confusion matrix for ARA-CNN trained with patches sized 74 µm extracted from initial tissue annotations. The mean accuracy is 84.35%.

	Tumor [%]	**Stroma** [%]	**Mixed** [%]	**Immune** [%]	**Vessel** [%]	**Bronchi** [%]	**Necrosis** [%]	**Lung** [%]	**Background** [%]
Tumor	90.34	0.25	1.15	3.72	0.59	0.86	0.88	2.21	0.00
Stroma	4.17	85.37	2.66	0.19	2.36	0.77	3.40	1.08	0.00
Mixed	14.65	6.73	61.49	10.69	1.09	0.99	1.68	2.67	0.00
Immune	0.59	0.32	0.48	94.89	0.59	0.16	0.65	2.15	0.16
Vessel	1.21	11.66	1.41	0.25	70.70	2.36	1.51	10.80	0.10
Bronchi	9.93	2.37	2.63	2.09	2.73	68.85	0.83	10.58	0.00
Necrosis	0.58	0.80	0.42	0.83	1.25	0.11	95.50	0.51	0.00
Lung	0.32	0.10	0.04	0.01	0.38	0.20	0.02	98.31	0.61
Background	0.00	0.00	0.00	0.00	0.12	0.00	0.00	3.59	96.29

Table 4.2: Confusion matrix for ARA-CNN trained with patches sized 87 µm extracted from initial tissue annotations. The mean accuracy is 85.21%.

	Tumor [%]	**Stroma** [%]	**Mixed** [%]	**Immune** [%]	**Vessel** [%]	**Bronchi** [%]	**Necrosis** [%]	**Lung** [%]	**Background** [%]
Tumor	90.27	0.27	0.91	4.17	0.73	0.65	1.08	1.94	0.00
Stroma	4.56	84.31	2.06	0.06	4.69	0.56	3.31	0.44	0.00
Mixed	15.32	4.19	62.58	10.32	3.23	0.97	1.61	1.77	0.00
Immune	0.58	0.08	0.58	94.79	0.74	0.08	0.58	2.15	0.41
Vessel	0.80	9.73	1.47	0.20	75.07	1.60	1.67	9.40	0.07
Bronchi	8.32	1.42	2.84	2.53	2.68	69.21	0.63	12.32	0.05
Necrosis	0.79	0.52	0.23	0.47	0.77	0.09	96.82	0.32	0.00
Lung	0.53	0.88	0.08	0.13	1.19	0.45	0.20	95.78	0.76
Background	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.95	98.05

Table 4.3: Confusion matrix for ARA-CNN trained with patches sized 100 µm extracted from initial tissue annotations. The mean accuracy is 84.35%.

	Tumor [%]	**Stroma** [%]	**Mixed** [%]	**Immune** [%]	**Vessel** [%]	**Bronchi** [%]	**Necrosis** [%]	**Lung** [%]	**Background** [%]
Tumor	88.22	0.44	1.63	3.89	0.59	1.33	1.56	2.33	0.00
Stroma	6.58	80.17	2.05	0.34	3.68	1.37	4.87	0.94	0.00
Mixed	12.83	4.78	60.65	11.52	1.74	1.74	3.48	3.26	0.00
Immune	0.88	0.25	0.63	95.38	1.00	0.00	0.50	1.38	0.00
Vessel	1.43	9.81	0.76	0.29	74.19	1.62	2.57	9.14	0.19
Bronchi	9.04	1.76	1.76	2.35	2.65	67.87	0.96	13.60	0.00
Necrosis	0.60	0.40	0.23	0.63	0.80	0.13	97.00	0.20	0.00
Lung	0.36	0.59	0.02	0.13	0.43	0.27	0.34	97.15	0.70
Background	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.47	98.53

4.4.3. Identification of TME spatial composition features in TCGA slides

We then sought to apply our trained ARA-CNN model to study the spatial architecture of the TME in H&E images from 411 LUAD patients downloaded from the TCGA database. Due to the fact that LUAD is more affected by genetic alterations, we focused the further analysis on this particular sub-type of lung cancer, omitting LUSC. We split each image into 87x87 µm patches and then normalized each patch to the same color space as the images in the *LubLung dataset* (see **4.3.2**). We used each patch as input to our ARA-CNN model, which returned the probabilities of assigning each patch to one of the nine tissue classes. We then segmented each image by assigning the most probable class to each patch (see **4.3.3**). For each image, we computed two sets of human-interpretable features that reflect the spatial structure of the TME: TIP and TMEC (see **4.3.4** and **Figure 4.6a**).

Across the investigated tissue classes, *tumor* and *lung* classes dominated the entire tissue composition, with a median t_{TUMOR} of 0.36 and a median t_{LUNG} of 0.22 (**Figure 4.6b**). The next three most abundant classes in the LUAD slides were *mixed*, *immune* and *bronchi* (with median prevalence of around 0.07). Finally, the least abundant classes were *stroma*, *vessel*, and *necrosis*. The most dominant classes of the tumor microenvironment were *lung* (median $m_{LUNG} = 0.32$), *immune* (median $m_{IMMUNE} = 0.18$) and *mixed* (median $m_{MIXED} = 0.17$). These classes were followed by *bronchi* (median $m_{BRONCHI} = 0.11$). The least abundant in the tumor microenvironment were *stroma*, *vessel* and *necrosis* classes. This indicates that in many patients, the tumor is surrounded by normal lung tissue and is confronted with an immune response. The abundance of all features, however, showed large variability across the

analyzed TCGA slides, indicating high heterogeneity of both the entire tissue and the tumor microenvironment composition.

4.4.4. TME features are predictive of patient survival

We then explored if our spatial features can be used to predict patient survival, given that the composition of the TME has been previously shown to influence disease aggressiveness and survival in various cancer types [82, 198]. To this end, we first stratified the 411 LUAD patients into two groups based on their TIP and TMEC feature levels (High vs. Low). The stratification was performed using the *survminer* R package, which selects the cut-off point between high and low values based on the significance to the survival outcome. Specifically, the method implements a test of independence of a response variable and the given feature using maximally selected rank statistics. For each feature, we compared survival between the two groups using the Kaplan-Meier estimator. Six TIP features (*vessel* $p = 0.0016$, *immune* $p = 0.0058$, *necrosis* $p = 0.0001$, *stroma* $p = 0.0352$, *bronchi* $p = 0.0079$ and *mixed* $p = 0.0040$) and five TMEC features (*vessel* $p = 0.0001$, *immune* $p = 0.0045$, *necrosis* $p = 0.0009$, *stroma* $p = 0.0086$, and *bronchi* $p = 0.0254$) showed statistically significant ($p < 0.05$, log rank test, two-sided) differences in survival between High and Low groups (**Figure 4.7a-k**).

To systematically assess the added value of the TIP or TMEC and to compare them to other predictive features, we trained several versions of a multivariate Cox proportional hazards model of the death hazard for the analyzed LUAD patients and assessed the performance of each model with Harrell's c-index [188]. The versions in question were based on different combinations of input features (see **4.3.5**). The best performing model yielded a median c-index of 0.723 and included clinical data (age, sex, pathologic stage, and smoking status), *EGFR*, *STK11* and *TP53* gene mutations, as well as TIP features. Inclusion of TMEC instead of TIP features yielded the second best model, with a slightly lower, but still high c-index of 0.709 (**Figure 4.7l**). All other models – including those based on spatial diversity metrics such as Shannon index [186], Simpson index [185] and ITLR [39] – resulted in lower c-index values. These results indicate that the TIP and TMEC features, which respectively reflect the repertoire of different tissues and their proportions across the entire examined tissue and across the TME, are superior to other spatial metrics in predicting patient survival. Arguably, as features, they bring more interpretable insight than spatial metrics such as ITLR, as well

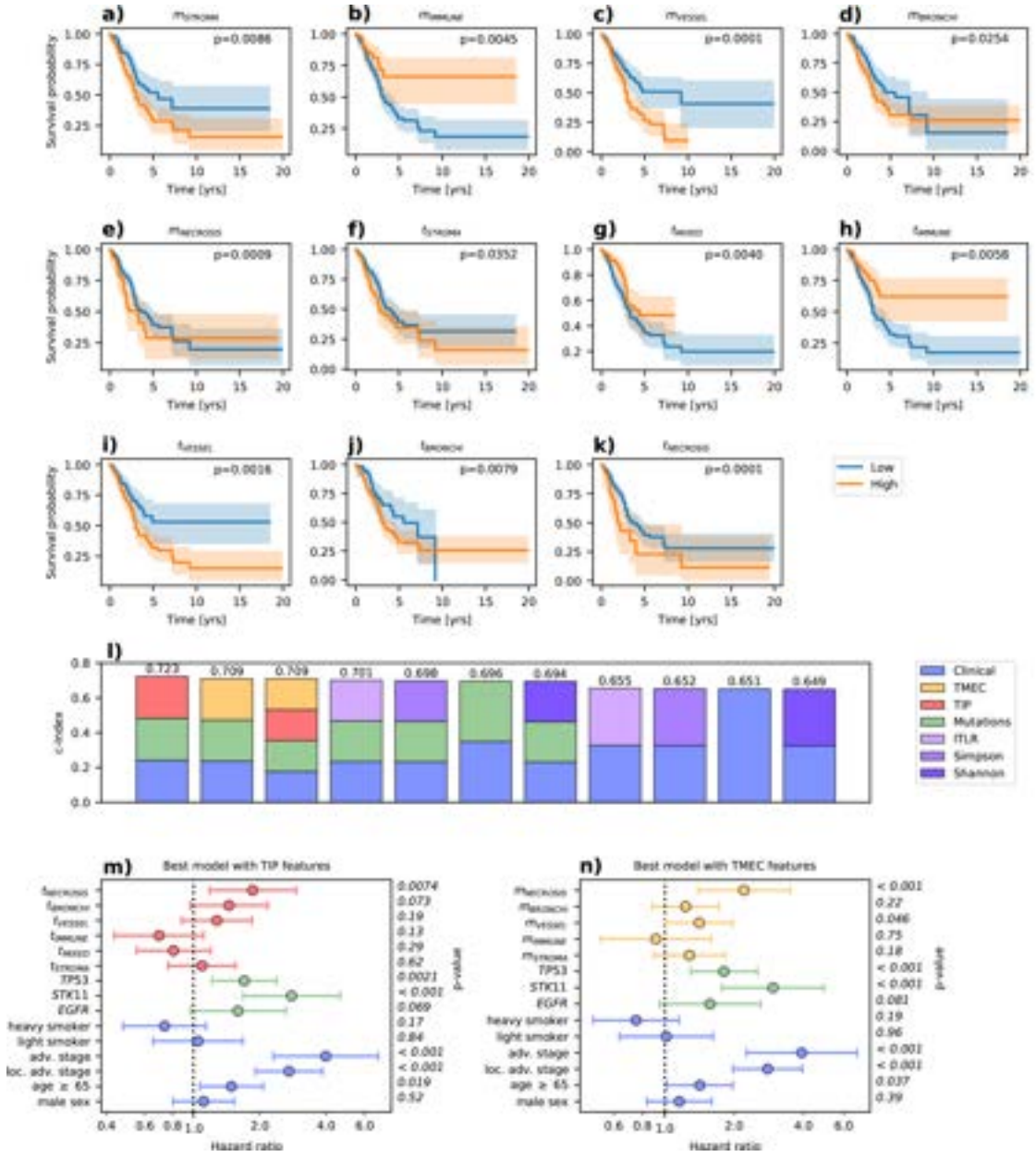

Figure 4.7: Survival prediction results. **(a-k)** Kaplan-Meier plots for TIP and TMEC features that result in patient stratification into two groups: with high and low values of the feature. Only features with statistically significant differences in patient survival are shown, as measured using the log rank test (p-values in the top right corner). The results correlate with previous studies of the relationship between these features and patient survival. **(l)** c-index scores for Cox models from survival prediction experiments performed with different feature sets. The best results were obtained for models with such feature sets that included TIP and TMEC features. **(m)** Hazard ratios for the best model that utilized the TIP features. The prevalence of the necrosis tissue class in the whole slide has a statistically significant negative effect on survival. **(n)** Hazard ratios for the best model that utilized the TMEC features. The presence of the necrosis tissue class and the vessel tissue class in the TME has a statistically significant negative effect on survival.

as the Shannon and Simpson indices, which summarize the proportions into a single measure. On top of that, given that the next best performing metric, ITLR, is computed based only on the abundance of the immune cells that neighbor the tumor, we conclude that other recognizable tissue types in the TME are also important for patient survival. This also agrees with the univariate Kaplan-Meier-based analysis (**Figure 4.7a-k**).

Next, we inspected the two best performing Cox models for the association between TIP and TMEC features and the death hazard accounting for the context of other features. A hazard ratio of 1 for a given feature indicates that the feature has no effect on survival, whereas a feature with hazard ratio larger than 1 indicates an increased death hazard and, therefore, a negative impact on survival. According to the best performing model, high abundances of $t_{NECROSIS}$ and t_{VESSEL} features in the H&E image were associated with increased hazard. Similarly, abundance of $t_{BRONCHI}$ and t_{STROMA} features had a negative effect on survival (**Figure 4.7m**). In contrast, t_{IMMUNE} and t_{MIXED} features were associated with a decreased death hazard and therefore longer survival (**Figure 4.7m**), in line with the established role of the immune system as a barrier against tumor progression [82, 198, 98]. Among mutation features, *TP53* and *STK11* mutations significantly increased ($p < 0.05$, Wald test, two-sided) the death hazard, in agreement with the results of the independent Kaplan-Meier analysis (**Figure 4.8**). The second best model, trained with TMEC instead of TIP features, yielded very similar results (**Figure 4.7n**).

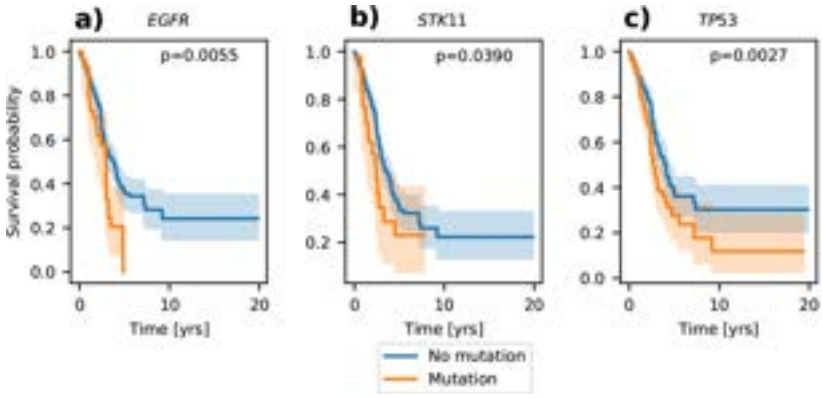

Figure 4.8: Kaplan-Meier plots for *EGFR*, *STK11* and *TP53* genes, stratified into patients with mutation and patients without mutation. The p-values were measured using the log rank test.

From among clinical features, statistically significant association was found for the advanced stage, locally advanced stage and age ≥ 65 features (Wald test p-value < 0.05). As expected, both for patients with locally advanced stage and with advanced stage the hazard

was increased compared to an early stage, with the advanced stage having the largest estimated hazard ratio of all the stages. Similarly, as expected, for older patients the hazard was larger. The estimated hazard ratio for light smokers was increased as compared to the non-smokers. Seemingly counter-intuitively, the hazard ratio for heavy smokers decreased. This may be explained by the fact that smoking is a well known risk factor for morbidity in lung cancer, but not mortality. In fact, patients who smoke after lung cancer diagnosis have a worse prognosis than patients who quit smoking. However, patients who smoke heavily before diagnosis may have a high tumor mutation burden, a high number of tumor antigens, and an active immune system with higher sensitivity to immune checkpoint inhibitors, which may prove beneficial for overall survival [199].

4.4.5. TME features are predictive of disease-relevant mutations

Mutation classification results

Next, we sought to investigate the association of the human-interpretable spatial composition features of H&E images with mutations in lung cancer genes. To this end, we trained classifiers for the mutation status of 13 genes that are frequently mutated in LUAD (see **4.2.5**). We evaluated eight different feature sets (see **4.3.6**) with two machine learning algorithms: logistic regression and random forest. Out of all 104 feature set and gene combinations, logistic regression was the better performing algorithm in 55 cases, while random forest performed better in the remaining 47 cases, indicating that for some genes non-linear relationships between the predictive features may be relevant for prediction of their mutations (**Table 4.4**). For 8 out of 13 considered genes (namely, *RET*, *KRAS*, *KEAP1*, *TP53*, *BRAF*, *PDGFRB*, *ROS1*, *STK11*), using the TIP or TMEC features gave the best result. For the next group of genes (*MET*, *ALK*, *DDR2*, *PIK3CA*), the best AUC was reached for models that utilized one of previously existing spatial metrics as features. Finally, for the *EGFR* gene, the clinical features alone were most predictive of its mutations.

The best AUC (73.5%) was reached for the *PDGFRB* gene mutation by a classifier using clinical data and both TIP and TMEC as features (**Table 4.4**). The best model without TIP and TMEC, and with the Simpson metric as a feature, yielded an AUC smaller by 3.4 p.p.. This shows that for the *PDGFRB* gene mutation, the full information about tissue distribution, not reduced to a single value using entropy and without focusing only on selected tissues, is highly relevant for its mutation status. The classification performance of the best

Table 4.4: Mutation/rearrangement classification AUC scores (given as % of area under the precision-recall curve ± standard deviation, averaged over 100 10-fold cross-validation experiments) for TCGA LUAD patients. The best result for each gene is marked in bold. In cases where the random forest classifier gave the best result, the cells are colored in yellow. Otherwise, if logistic regression gave the best result, the cells are colored in light blue.

Gene	Mutation count	Clinical	Clinical + ITLR	Clinical + Shannon	Clinical + Simpson	Clinical + TMEC	Clinical + TIP	Clinical + TMEC + TIP	TMEC
RET	25	53.70 ± 2.21	50.76 ± 2.39	58.03 ± 2.13	57.76 ± 2.09	**67.14 ± 2.75**	59.46 ± 2.13	65.32 ± 1.95	64.39 ± 2.73
KRAS	96	61.12 ± 0.88	61.36 ± 0.84	60.06 ± 1.04	60.51 ± 0.99	**62.37 ± 1.50**	62.07 ± 1.66	60.98 ± 1.71	55.58 ± 1.54
KEAP1	96	51.35 ± 1.12	53.98 ± 1.60	51.37 ± 1.17	50.87 ± 1.21	**61.07 ± 1.60**	60.33 ± 1.85	60.05 ± 1.58	56.85 ± 1.48
TP53	200	55.28 ± 0.74	54.38 ± 0.75	55.18 ± 0.78	55.02 ± 0.79	**58.59 ± 1.30**	57.87 ± 1.26	57.78 ± 1.52	57.14 ± 1.48
BRAF	30	55.38 ± 1.74	55.14 ± 2.49	56.59 ± 1.76	56.87 ± 1.74	55.26 ± 2.66	**57.10 ± 1.95**	56.23 ± 2.14	51.35 ± 2.88
PDGFRB	15	67.85 ± 2.51	69.75 ± 2.79	69.91 ± 3.23	70.06 ± 2.97	72.03 ± 2.04	72.78 ± 2.48	**73.50 ± 2.02**	52.13 ± 3.83
ROS1	18	58.29 ± 3.58	60.74 ± 2.76	51.27 ± 3.20	49.97 ± 3.20	59.44 ± 4.69	60.07 ± 2.55	**61.45 ± 4.54**	59.58 ± 4.08
STK11	44	53.98 ± 1.76	53.91 ± 1.67	53.56 ± 1.83	61.39 ± 2.22	58.41 ± 1.51	54.89 ± 2.64	59.22 ± 2.30	**62.89 ± 2.51**
MET	19	61.22 ± 2.31	**69.20 ± 2.31**	63.14 ± 2.39	64.19 ± 2.35	66.58 ± 3.70	68.10 ± 2.45	63.68 ± 2.76	63.36 ± 4.28
ALK	28	53.55 ± 2.23	48.42 ± 2.49	**62.64 ± 3.05**	59.51 ± 2.23	50.32 ± 2.88	61.53 ± 1.70	58.71 ± 1.95	49.41 ± 2.89
DDR2	18	57.17 ± 3.81	56.89 ± 4.21	**57.38 ± 3.80**	56.69 ± 3.82	55.63 ± 4.30	54.65 ± 3.96	52.56 ± 4.37	51.78 ± 5.13
PIK3CA	23	49.14 ± 1.78	51.30 ± 2.67	51.07 ± 2.11	**58.35 ± 2.78**	53.90 ± 3.27	56.96 ± 3.46	58.01 ± 3.18	56.15 ± 3.22
EGFR	49	**65.09 ± 1.50**	61.48 ± 2.49	63.32 ± 1.25	64.90 ± 1.97	59.01 ± 1.60	59.69 ± 1.56	57.93 ± 1.60	49.05 ± 1.97

model using both TIP and TMEC for that gene is only slightly smaller than AUC of 75%, as previously reported for a deep learning model trained on raw H&E images [111], but is less difficult to interpret. For eight other genes (*RET*, *KRAS*, *KEAP1*, *ROS1*, *STK11*, *MET*, *ALK*, *EGFR*), the best AUC ranged between 60% and 70%, while for the four remaining ones (*TP53*, *BRAF*, *DDR2*, *PIK3CA*) the best AUC ranged between 55% and 60%. For some of the genes, the inclusion of TIP or TMEC features resulted in impressive improvements compared to other feature sets. For *RET*, the model trained with clinical data and TMEC outperformed the best model without TIP and TMEC features, but including the Shannon metric, by around 9.1 p.p.. Similarly, for *KEAP1* the classification performance increased by 7 p.p. compared to models without TIP or TMEC. These results indicate that, in LUAD, there exists a subset of tumor mutations that significantly correlate with how the TME is structured, and that both TIP and TMEC features are predictive of the presence of these mutations.

Feature importance in mutation classification

We then inspected the two best performing models in the mutation classification task that utilized TIP and TMEC features to find which predictor features were the most important for identifying mutations. Both of the algorithms used – logistic regression and random forest

– are easily interpretable because they allow effective identification of the most important features. First, we analyzed the logistic regression classifier of *PDGFRB* mutations with clinical, TMEC and TIP features (**Figure 4.9a**). The most important features positively correlated with *PDGFRB* mutation were sex, m_{MIXED} – corresponding to the proportion of the mixed tissue in the TME – and t_{TUMOR} – corresponding to the fraction of the entire slide occupied by the tumor. On the other hand, the most negatively correlated (i.e., decreasing the chance of mutation) features were non-smoker status, t_{IMMUNE}, and $m_{BRONCHI}$. Next, we inspected the random forest classifier of *RET* mutations, which included clinical and TMEC features in its feature set (**Figure 4.9b**). The latter proved to be of larger importance than the former ones. Indeed, *RET* mutations were found to be most associated with the prevalence of different tissues in the tumor microenvironment, with *bronchi* and *vessel* identified as the most impactful tissues, followed by *mixed*, *stroma*, *lung*, *immune* and *necrosis*. This observation might be explained by the fact that, in LUAD, *RET* mutations mainly consist of rearrangements between *RET* gene and its common fusion partners such as *KIF5B*, *CCDC6*, *CUX1*, *TRIM33*, *NCOA4*, *KIAA1468* and *KIAA1217* genes.

In addition to feature importance, we also inspected the distributions of the values of the TIP and TMEC spatial composition features for patients with and without mutations of the *PDGFRB* and *RET* genes. For both of them, we selected the four most important TIP or TMEC features and assessed their value distributions separately for mutated and non-mutated cases. For *PDGFRB*, these features were: m_{MIXED} and $m_{BRONCHI}$ (TMEC features), as well as t_{VESSEL} and t_{IMMUNE} (TIP features) (**Figure 4.9c**). We detected a statistically significant difference between the value distributions (two-sided Wilcoxon test p-value < 0.05) for t_{VESSEL}. For *RET*, the four most important features were TMEC features $m_{BRONCHI}$, m_{MIXED}, m_{VESSEL} and m_{STROMA}, with m_{MIXED} and $m_{BRONCHI}$ features having a statistically significant difference in value distributions between mutated and non-mutated tumors (**Figure 4.9d**). These results indicate that the spatial composition features TIP and TMEC are different between tumors with and without *PDGFRB* and *RET* mutations, and their importance for the classification of mutations of these genes is not incidental.

4.4.6. Tissue segmentation with CRFs

The results presented so far are strictly correlated with the quality of input segmentations used to calculate TIP and TMEC features. If these segmentations were to be of poor quality,

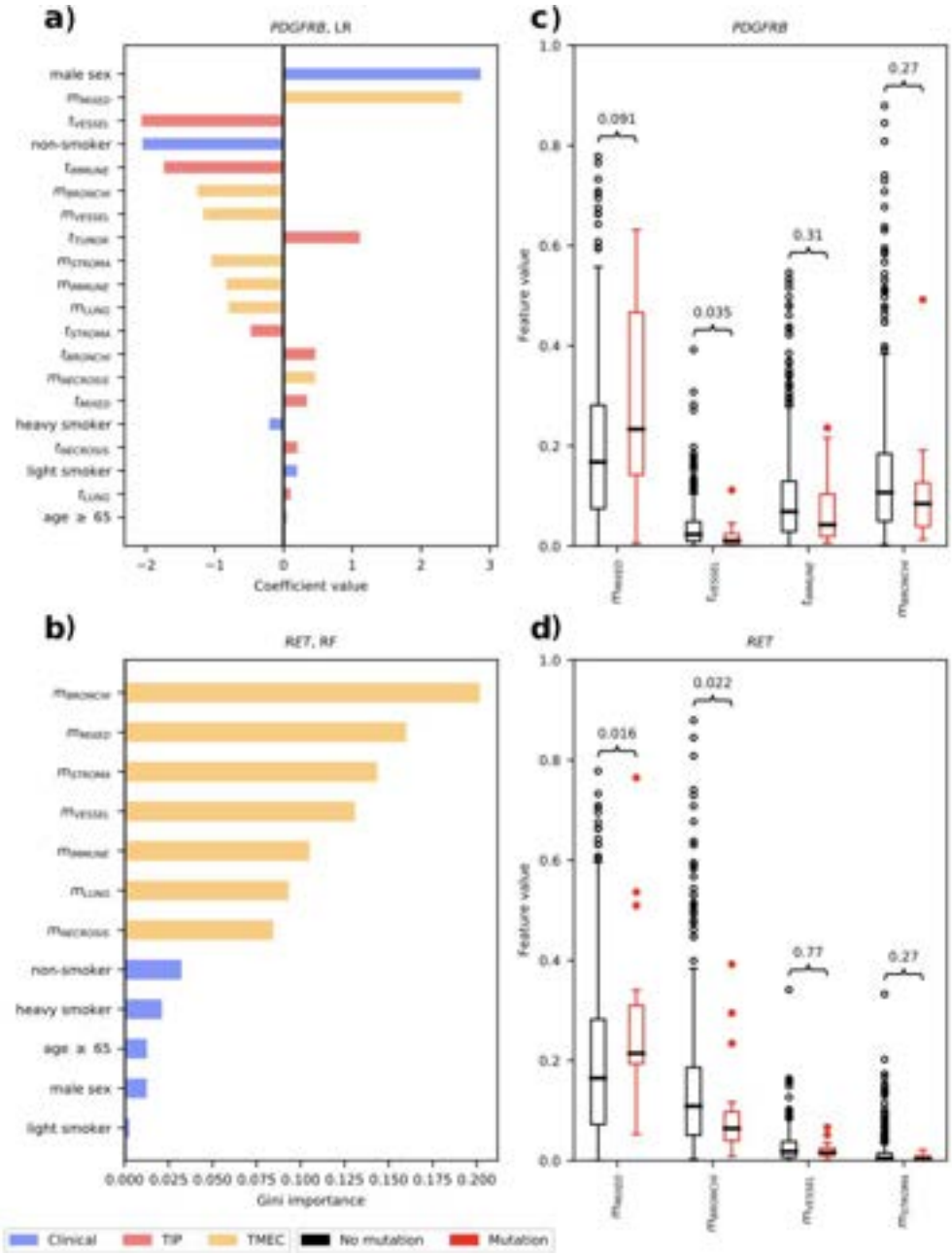

Figure 4.9: Feature importance for the two best performing mutation classification models that utilized TIP and TMEC features. **(a)** Feature importance for the PDGFRB gene mutation classifier (logistic regression). Here, feature importance is measured by the value of its regression coefficient. **(b)** Feature importance for the RET gene mutation classifier (random forest). Here, the importance is measured by the reduction of the Gini index obtained when the feature is added to the tree, averaged across the trees in the random forest model. **(c)** Distribution of feature values for four of the most important TIP or TMEC features, as presented in (a), divided between patients with the mutated and non-mutated PDGFRB gene. **(d)** Distribution of feature values for four of the most important TMEC features, as presented in (b), divided between patients with the mutated and non-mutated RET gene.

i.e. they did not represent the TME correctly, then spatial features would have inherent error in them. The segmentations produced by ARA-CNN were assessed by a pathologist, who deemed them as satisfactory, but many mistakes were still present. The most prominent one was the problem of unrelated tissue patches dotted inside contiguous regions of another tissue (**Figure 4.10a**). This problem arises because ARA-CNN does not take the patch context into account during classification, as each patch is considered separately. Such errors disturb the TMEC metric, because incorrect neighbors are added to the total count (see **4.3.4**). To mitigate this issue, we developed a post-processing model meant to correct segmentation mistakes. This model is based on the CRF method (see **4.3.7-4.3.9**).

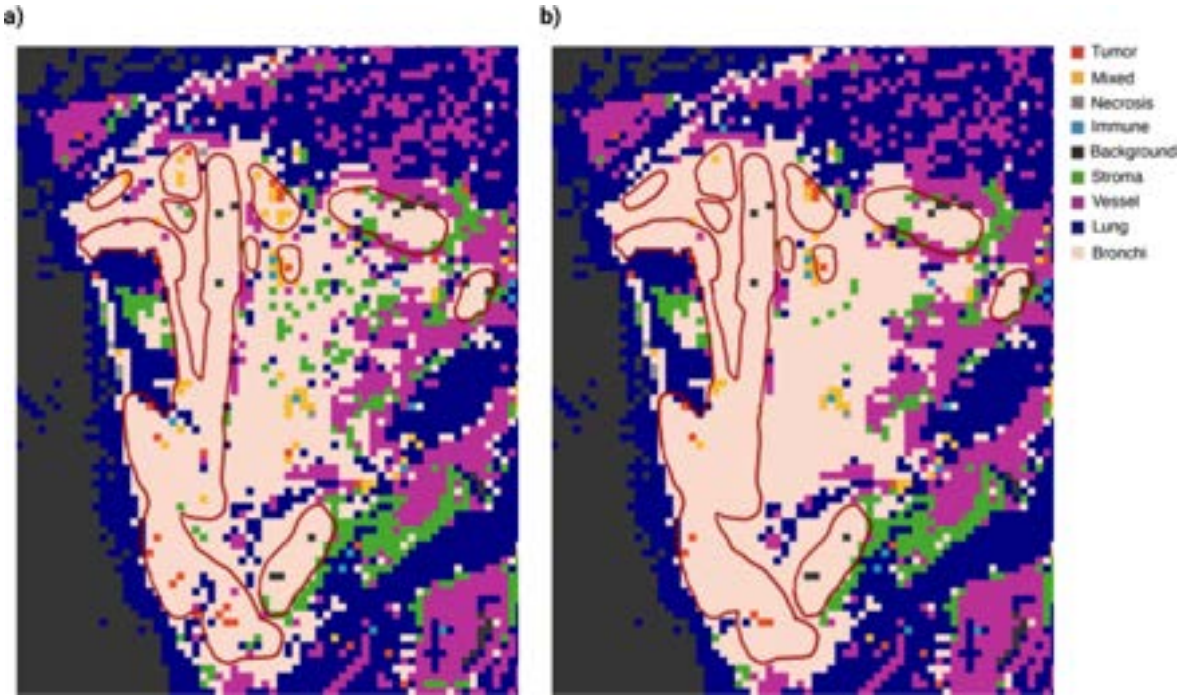

Figure 4.10: Visual segmentation comparison between ARA-CNN and ARA-CNN + CRF. The dark red outlines denote ground truth regions of the *bronchi* class annotated by the pathologist. **(a)** Region of slide 11 segmented with ARA-CNN. Unrelated tissue patches are dotted inside regions of *bronchi* tissue. **(b)** Region of slide 11 segmented with ARA-CNN + CRF. It is clearly visible that areas of *bronchi* tissue are much more contiguous.

The CRF-based post-processing improves segmentation results. When segmentations were compared on a per-slide basis (**Table 4.5**), the combined ARA-CNN + CRF setup achieved better results than ARA-CNN without any post-processing for 18 out of 24 considered slides, worse for 1 slide and identical for 5 slides. The improvement ranged from 0.04 p.p. to 10.23 p.p., so for some slides the contiguous tissue areas were made consider-

ably more homogeneous. This is further confirmed upon analyzing the segmentation results visually (**Figure 4.10b**). It is evident that contiguous regions became more consistent after applying CRF post-processing. On average, adding this form of post-processing improved the per-slide accuracy by 1.07 p.p.. These results prove the effectiveness of the proposed approach.

Table 4.5: Comparison of segmentation accuracy between ARA-CNN and ARA-CNN + CRF on a per-slide basis. The slides were taken from *LubLung* and limited to those with tissue regions annotated by a pathologist. The green cells indicate a better result.

Slide ID	ARA-CNN accuracy [%]	ARA-CNN + CRF accuracy [%]
1	93.08	93.76
2	87.18	87.18
3	79.15	81.42
4	83.36	83.40
5	90.20	91.28
6	91.09	91.09
7	93.03	94.06
8	52.44	52.72
9	49.46	49.80
10	92.42	92.66
11	86.99	87.25
12	92.90	92.94
13	89.78	91.84
15	87.83	89.96
18	92.64	93.11
19	87.10	85.24
20	68.35	72.92
22	78.21	78.60
23	68.75	68.75
24	74.10	84.33
25	85.79	86.44
27	95.94	96.75
32	61.23	61.23
49	85.02	85.02

Furthermore, we also computed confusion matrices of overall per-class accuracy for all considered slides (**Figure 4.11**). The ARA-CNN + CRF approach improved accuracy for *tumor*, *stroma*, *mixed*, *immune*, *vessel* and *bronchi* classes, where the highest improvement of 2 p.p. was observed for the *vessel* class. At the same time, the new approach lowered accuracy for *background*, *necrosis* and *lung* classes. This is most likely due to the slicing procedure - if there were many slices with different patterns of these classes together, then the model would

not have been able to learn them properly. Overall, the mean per-class accuracy improved by 0.3 p.p. for the ARA-CNN + CRF approach.

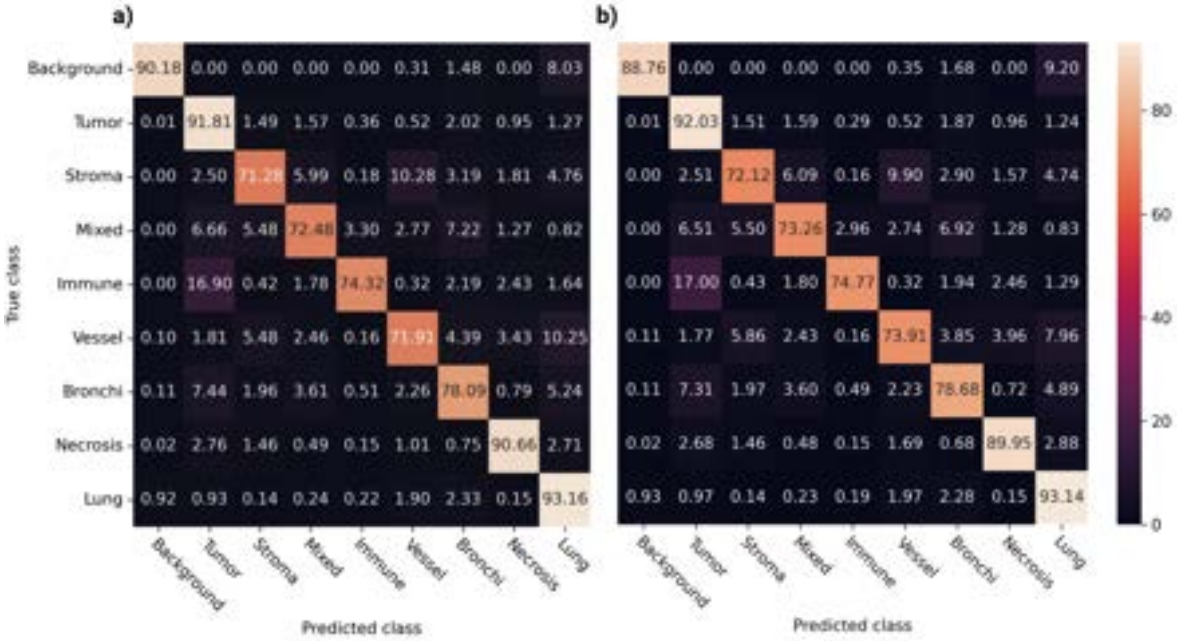

Figure 4.11: Comparison of segmentation accuracy between ARA-CNN and ARA-CNN + CRF on a per-class basis. **(a)** Confusion matrix for ARA-CNN without any post-processing. **(b)** Confusion matrix for ARA-CNN with CRF post-processing applied. On average, the classification accuracy within annotated tissue regions increased by 0.3 p.p..

4.5. Conclusions

We have developed a novel H&E image classification model, ARA-CNN, and a training dataset of annotated tissue patches from LUAD and LUSC H&E images, *LubLung*. Both considerably expand the current ability to analyze the TME automatically and quantitatively in lung cancer samples, which in turn has important implications for patient stratification and precision treatment. TIP and TMEC features, which we have introduced in this work, provide a novel way of capturing the composition and spatial structure of the TME, and are predictive of both overall survival and clinically relevant mutations. Spatial statistics of H&E images in the form of metrics that quantify colocalization of cell or tissue types, have been previously shown to be predictive of patient survival [39]. However, these metrics are computed based on a limited number of features, such as counts of tumor and immune cells. Other approaches that try to link the structure of tumor tissue and TME with either gene mutations or patient survival

are end-to-end deep learning models and work as 'black boxes' [200, 106, 80, 201, 111, 107]. Instead, our approach allows explicit interpretability, as it decouples H&E slide inference from downstream tasks (i.e., mutation classification and survival analysis). The TIP and TMEC features are per se human interpretable, so it is possible to precisely pinpoint which tissue types are the most important. Our approach requires the initial tissue classification to be as accurate as possible. We ensured this to be the case by using ARA-CNN, which performs excellently in classifying nine tissue classes present in lung cancer H&E images. To foster further research in predictive spatial statistics based on a rich repertoire of segmented lung cancer tissues, in addition to *LubLung* we also shared the segmented TCGA images as a separate dataset, named *SegLungTCGA*.

We also experimented with further improving the segmentation results by applying a grid-based CRF post-processing. We trained this model on segmentations produced by ARA-CNN and ground truth annotations. We observed an improvement in classification metrics - on a per-slide basis the proposed post-processing increased the mean accuracy by 1.07 p.p., while on a per-class basis it was 0.3 p.p. instead. This shows that such methods can effectively correct tissue segmentations. However, the presented post-processing approach is quite limited, as it has to work with ARA-CNN output, which includes errors made by that model. There are several alternative approaches that could be explored in order to incorporate information about neighboring patches during tissue classification itself. One of them is the inclusion of the CRF model into an end-to-end training process together with ARA-CNN. In this setup, CRF would essentially become the last layer in a combined ARA-CNN-CRF network, which would be trained with a single loss objective. Such combined CNN+CRF architectures have been used in other fields before [202, 203, 204]. Another approach involves replacing a CNN-based architecture with another one, Vision Transformer [205]. This type of model is gaining traction in recent time, as it is inspired by the Transformer architecture, which became ubiquitous in the field of natural language processing. In short, Vision Transformer encodes a visual encoding based on sequences of image patches. Thus, it inherently works in a spatial manner, learning about neighborhoods of objects in the training data. As such, it may be a natural fit for the problem of tissue segmentation in H&E slides.

Our analysis revealed that patient stratification based on TIP and TMEC features yields significant differences in patient survival between the strata. Moreover, the most predictive survival models included TIP and TMEC features. These findings are supported by previous clinical studies. It has been shown that blood vessel invasion is a major prognostic factor in

lung cancer survival [206, 93]. Similarly, there have been studies which proved that tumor necrosis is a significant risk factor for survival in lung cancer [207]. However, the complexity of the entire lung microenvironment plays a key role in the development of primary lung carcinomas and offers a resource of targets for personalized therapy development. Targeting the angiogenesis and immune cells has elucidated the prognostic and pathophysiological roles of other components of the TME in lung cancer [92, 208]. In the end, the combination of the clinical and genetic information with the TME landscape may play a pivotal role in predicting the type and duration of response to personalized therapies.

We found eight genes relevant to lung cancer (*PDGFRB*, *RET*, *KRAS*, *KEAP1*, *ROS1*, *STK11*, *MET* and *ALK*), for which integrating clinical data with our TME features clearly improves the ability to predict mutations in these genes. We speculate that mutations of these genes may alter cellular interactions, and hence the spatial arrangement of the TME visible in H&E images. For *RET*, *ROS1* and *ALK* genes, mutations mainly consist of chromosomal rearrangements which produce chimeric proteins that might affect the cellular organisation within the TME [209, 210, 211]. Likewise, loss of *STK11/LKB1* overlapping with oncogenic *KRAS* mutations is associated with increased neutrophil recruitment, and decreased T-cells infiltration in lung cancer tumors [212]. Moreover, *STK11* mutations often coexist with *KEAP1* mutations that relate to cellular resistance to oxidative stress [213], and co-occurrence of *KEAP1* mutations and *PTEN* inactivation is an indicator of an immunologically "cold" tumor [214]. We speculate that each of these mutations might slightly affect the cellular morphology in H&E images in a way that is not apparent to the human eye, but can be captured by deep-learning algorithms.

Our findings concern mutations of clinically relevant genes, and as such may have clinical implications. For example, both *RET* and *PDGFRB* are clinically relevant LUAD cancer genes. *RET* has proto-oncogene properties and its fusions, which occur in 1–2% of LUAD [215], are associated with a high risk of brain metastasis [216]. However, last clinical trials indicated that they may be effectively targeted by *RET* tyrosine kinase inhibitors such as pralsetinib and selpercatinib [215]. *PDGFRB* is a member of the PDGF/PDGFR axis that is recognized as a key regulator of mesenchymal cell activity in TME [217], and several new agents (linifanib, motesanib, olaratumab) that block the PDGFR signaling are being tested in LUAD [218]. In breast, colon, pancreas and prostate cancers, the high stromal expression of the PDGFRβ protein has been associated with poor prognosis [218], however its prognostic relevance in tumors of epithelial origin is inconclusive [217]. It was only confirmed

that a relative expression of PDGFRs is a strong and independent predictor of longer survival for surgical stages of lung cancer (I-IIIA) [218].

Our approach has several limitations. In the mutation classification tasks, we used simple machine learning models – logistic regression and random forest. An end-to-end deep learning model might give better results, however, as discussed above, these models suffer when it comes to interpretability. Another limitation is the fact that ARA-CNN works on a patch-based basis. An alternative to that is a cell-based classifier, which could produce more fine-grained segmentations and in turn enable a more precise computation of spatial statistics. On the other hand, with a patch-based approach, a suitably small patch size can be selected, as we did in this study. Such small patches can be assumed to be homogeneous when it comes to cell types and can enable a precise computation of summary statistics such as the TIP and TMEC features, that we have introduced here. However, due to the aggregation-based nature of these features, pathological events such as angioinvasions are hard to model properly. This is consistent with human-level perception of H&E images, as angioinvasions are hard to assess even for pathologists. Furthermore, in the survival prediction task we did not have access to and hence did not utilize the treatment information as features. Since treatment has a large impact on survival, using this data is expected to improve prediction performance by a large margin. However, the main focus of our study was not to deliver the best performing survival prediction approach, but rather to assess the predictive power of our proposed spatial composition features, which can be performed without including the treatment data in the model. Lastly, to apply our approach to another cancer type, one would need to retrain the ARA-CNN model, which necessitates substantial input from a trained pathologist. This training effort can be minimized by utilizing the active learning component of ARA, which shortens the number of iterations required to build an effective training dataset. For colorectal cancer, a pre-trained model is available from a previous study [123] (Chapter 3).

Compared to data from antibody-based methods for multiplex cancer tissue imaging analyzed by several recent studies [219, 220, 221, 222], the data analyzed in this work is limited in terms of the number of cell types and their states it allows to identify. In particular, many types of immune cells are not distinguishable in H&E slides even by expert pathologists. In contrast to multiplexed antibody imaging data, however, H&E slides are abundant and routinely used in the clinic and are becoming more commonly digitized. Thus, predictive models operating on H&E data are more likely to be adopted in clinical practice.

The analysis presented here shows that there is a correspondence between the spatial

structure in H&E images for LUAD and both gene mutations and patient survival. Not every mutation is expected to have an effect on tissue prevalence or tumor neighborhood structure, so it is not surprising that for some of the analyzed genes the mutation classification performance did not exceed an AUC of 0.6. In contrast, it is striking that there are genes for which adding tissue composition data to the clinical information improves classification results. Finally, it is also surprising that our TIP and TMEC features, as well as other metrics of TME spatial organization, such as ITLR, can give good results in terms of both mutation classification and survival analysis.

Chapter 5

Summary

This dissertation presented a spectrum of computational methods for image analysis, with a focus on the study of informative image features in both traditional computer vision and deep learning systems. These methods were applied empirically primarily for the analysis of the tumor microenvironment in histopathological images.

First, in Chapter 2 we presented a study of traditional hand-engineered image features and their application to the problem of visual product recommendations in e-commerce. We developed and deployed an effective visual recommendation system using existing open-source tools. Additionally, we tested the performance of several known low-level image features in a series of automatic and user-centric tests. The results obtained here were published in proceedings of SIGIR 2016 [122]. Based on them and ongoing developments in computer vision, we decided to pursue further study in the field of deep learning.

Second, in Chapter 3 we discussed a new Bayesian deep learning framework for accurate, reliable and active classification of up to eight tissue types in histopathological images of colorectal cancer. We showed that the classification accuracy of the model within this framework, ARA-CNN, exceeds results achieved by other methods that used the same training dataset [65] as us. ARA-CNN achieved an almost perfect error rate of 0.89% in binary classification and a best in class error rate of 7.56% in a full 8-class classification task. Additionally, thanks to the Bayesian nature of the model, we were able to measure the uncertainty of each prediction, which allowed us to study the utility of uncertainty evaluation methods in several important applications. First, we showed that the Entropy H uncertainty measure is an effective acquisition function in active learning, as it shortened the training time needed to achieve maximum accuracy by 45% compared to random sampling. Next, we proved that

the same H measure can be successfully applied in an important problem of identifying mislabeled training samples. Finally, we showed empirically that highly uncertain tissue patches are hard to distinguish even for a trained pathologist, while highly certain ones are very easy to classify. Overall, these results show high effectiveness and usefulness of the ARA framework as a whole and the ARA-CNN model in particular in the field of digital pathology. This study was published and the ARA-CNN model was made available on GitHub [123].

Next, in Chapter 4 we discussed applying ARA to a different cancer type - lung cancer. We created a new dataset of lung cancer tissue patches, *LubLung*, utilizing a human-in-the-loop active learning process. We trained ARA-CNN with this new dataset and achieved a very good mean AUC of 0.94 in a 9-class classification task. We then applied the trained model to H&E slides from lung cancer patients stored in the TCGA database. The segmented slides were quantified with two new spatial features: TIP and TMEC. We then used these features to successfully model patient survival (c-index up to 0.723) and predict gene mutations (AUC up to 73.5% for *PDGFRB*) in the studied TCGA patient cohort. Finally, we also presented a post-processing method with a grid-based CRF model, which improved initial ARA-CNN segmentation accuracy by 1.07 p.p. (mean per-slide accuracy). The presented approach can provide important insights for designing novel cancer treatments through linking the spatial structure of the tumor microenvironment in LUAD to gene mutations and patient survival. It can also expand our understanding of the effects that the tumor microenvironment has on tumor evolutionary processes. The presented framework is generalizable, so it can be extended to other tumor types. We therefore envision that, in the future, our quantitative approach will become incorporated in routine diagnostics for LUAD and other cancer types. This work was published in a pre-print paper [124] and is currently submitted for publication in a peer-reviewed journal. Both *LubLung* and a new dataset of segmented TCGA lung cancer tissue slides, *SegLungTCGA*, were made publicly available on GitHub.

All in all, the computational methods presented in this dissertation represent a significant contribution to the field of digital pathology. It is clear that traditional hand-crafted image features, while useful in some applications (such as the one presented in Chapter 2), do not have sufficient semantic capacity to fully capture all complexities present in H&E images. In contrast, the ARA framework and its further extension in the form of TIP and TMEC features are capable of encoding the structure of the tumor microenvironment and reasoning about it in several downstream tasks. Thus, the presented ARA framework has the potential to be utilized in medical practice by pathologists. Possible future research directions include

methodological advancements involving updates to the ARA-CNN architecture, exploration of Vision Transformer models for tissue segmentation, development of more spatial features in addition to TIP and TMEC and application of these features to more downstream tasks.

Bibliography

[1] Hubel, D. H. & Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology* **148**, 574–591 (1959). URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1363130/`.

[2] Hubel, D. H. & Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology* **160**, 106–154 (1962). URL `https://onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1962.sp006837`.

[3] Marr, D. *Vision: a computational investigation into the human representation and processing of visual information* (MIT Press, Cambridge, Mass, 2010).

[4] Kuruvilla, J., Sukumaran, D., Sankar, A. & Joy, S. P. A review on image processing and image segmentation. In *2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)*, 198–203 (2016).

[5] Barragán-Montero, A. *et al.* Artificial intelligence and machine learning for medical imaging: A technology review. *Physica Medica* **83**, 242–256 (2021). URL `https://www.sciencedirect.com/science/article/pii/S1120179721001733`.

[6] Rana, K. & Kaur, P. Review on Machine Learning Based Algorithms Used in Autonomous Cars. SSRN Scholarly Paper ID 3708283, Social Science Research Network, Rochester, NY (2018). URL `https://papers.ssrn.com/abstract=3708283`.

[7] Zakria *et al.* Trends in Vehicle Re-identification Past, Present, and Future: A Comprehensive Review. *arXiv:2102.09744 [cs]* (2021). URL `http://arxiv.org/abs/2102.09744`.

[8] Wang, L., Chen, W., Yang, W., Bi, F. & Yu, F. R. A State-of-the-Art Review on Image Synthesis With Generative Adversarial Networks. *IEEE Access* **8**, 63514–63537 (2020).

[9] Wang, S., Han, K. & Jin, J. Review of image low-level feature extraction methods for content-based image retrieval. *Sensor Review* **39**, 783–809 (2019). URL https://doi.org/10.1108/SR-04-2019-0092.

[10] Dubey, S. R. A Decade Survey of Content Based Image Retrieval using Deep Learning. *IEEE Transactions on Circuits and Systems for Video Technology* 1–1 (2021). URL http://arxiv.org/abs/2012.00641.

[11] Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Communications of the ACM* **60**, 84–90 (2017). URL https://dl.acm.org/doi/10.1145/3065386.

[12] Razavian, A. S., Azizpour, H., Sullivan, J. & Carlsson, S. CNN Features off-the-shelf: an Astounding Baseline for Recognition. *arXiv:1403.6382 [cs]* (2014). URL http://arxiv.org/abs/1403.6382.

[13] Tajbakhsh, N. *et al.* Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning? *IEEE Transactions on Medical Imaging* **35**, 1299–1312 (2016).

[14] He, K., Zhang, X., Ren, S. & Sun, J. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]* (2015). URL http://arxiv.org/abs/1512.03385.

[15] Kolen, J. F. & Kremer, S. C. Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies. In *A Field Guide to Dynamical Recurrent Networks*, 237–243 (IEEE, 2001). URL https://ieeexplore.ieee.org/document/5264952.

[16] Sundermeyer, M., Schlüter, R. & Ney, H. LSTM neural networks for language modeling. In *Interspeech 2012*, 194–197 (ISCA, 2012).

[17] Cho, K. *et al.* Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv:1406.1078 [cs, stat]* (2014). URL http://arxiv.org/abs/1406.1078.

[18] Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. Gated feedback recurrent neural networks. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, 2067–2075 (JMLR.org, Lille, France, 2015).

[19] Ioffe, S. & Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning*, 448–456 (PMLR, 2015). URL http://proceedings.mlr.press/v37/ioffe15.html.

[20] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* **15**, 1929–1958 (2014). URL http://jmlr.org/papers/v15/srivastava14a.html.

[21] Szegedy, C. *et al.* Intriguing properties of neural networks. *arXiv:1312.6199 [cs]* (2014). URL http://arxiv.org/abs/1312.6199.

[22] Jospin, L. V., Buntine, W., Boussaid, F., Laga, H. & Bennamoun, M. Hands-on Bayesian Neural Networks – a Tutorial for Deep Learning Users. *arXiv:2007.06823 [cs, stat]* (2022). URL http://arxiv.org/abs/2007.06823.

[23] Gal, Y. & Ghahramani, Z. Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference. *arXiv:1506.02158 [cs, stat]* (2016). URL http://arxiv.org/abs/1506.02158.

[24] Zhang, C., Butepage, J., Kjellstrom, H. & Mandt, S. Advances in Variational Inference. *arXiv:1711.05597 [cs, stat]* (2018). URL http://arxiv.org/abs/1711.05597.

[25] Li, Y. & Gal, Y. Dropout Inference in Bayesian Neural Networks with Alpha-divergences. *arXiv:1703.02914 [cs, stat]* (2017). URL http://arxiv.org/abs/1703.02914.

[26] Rawat, A., Wistuba, M. & Nicolae, M.-I. Adversarial Phenomenon in the Eyes of Bayesian Deep Learning. *arXiv:1711.08244 [cs, stat]* (2017). URL http://arxiv.org/abs/1711.08244.

[27] Feinman, R., Curtin, R. R., Shintre, S. & Gardner, A. B. Detecting Adversarial Samples from Artifacts. *arXiv:1703.00410 [cs, stat]* (2017). URL http://arxiv.org/abs/1703.00410.

[28] Kiureghian, A. D. & Ditlevsen, O. Aleatory or epistemic? Does it matter? *Structural Safety* **31**, 105–112 (2009). URL https://www.sciencedirect.com/science/article/pii/S0167473008000556.

[29] Brodley, C. E. & Friedl, M. A. Identifying Mislabeled Training Data. *Journal of Artificial Intelligence Research* **11**, 131–167 (1999). URL http://arxiv.org/abs/1106.0219.

[30] Hao, D., Zhang, L., Sumkin, J., Mohamed, A. & Wu, S. Inaccurate Labels in Weakly-Supervised Deep Learning: Automatic Identification and Correction and Their Impact on Classification Performance. *IEEE Journal of Biomedical and Health Informatics* **24**, 2701–2710 (2020).

[31] Sukhbaatar, S., Bruna, J., Paluri, M., Bourdev, L. & Fergus, R. Training Convolutional Networks with Noisy Labels. *arXiv:1406.2080 [cs]* (2015). URL http://arxiv.org/abs/1406.2080.

[32] Yang, Y. & Whinston, A. Identifying Mislabeled Images in Supervised Learning Utilizing Autoencoder. *arXiv:2011.03667 [cs]* (2021). URL http://arxiv.org/abs/2011.03667.

[33] Köhler, J. M., Autenrieth, M. & Beluch, W. H. Uncertainty Based Detection and Relabeling of Noisy Image Labels. *arXiv:1906.11876 [cs, stat]* (2019). URL http://arxiv.org/abs/1906.11876.

[34] Cohn, D. A., Ghahramani, Z. & Jordan, M. I. Active Learning with Statistical Models. *arXiv:cs/9603104* (1996). URL http://arxiv.org/abs/cs/9603104.

[35] Gal, Y., Islam, R. & Ghahramani, Z. Deep Bayesian Active Learning with Image Data. *arXiv:1703.02910 [cs, stat]* (2017). URL http://arxiv.org/abs/1703.02910.

[36] Tolimieri, N., Shelton, A., Feist, B. & Simon, V. Can we increase our confidence about the locations of biodiversity 'hotspots' by using multiple diversity indices? *Ecosphere* **6**, art290 (2015).

[37] Kitzes, J., Brush, M. & Walters, K. A unified framework for species spatial patterns: Linking the occupancy area curve, Taylor's Law, the neighborhood density function and two-plot species turnover. *Ecology Letters* **24**, 2043–2053 (2021). URL https://onlinelibrary.wiley.com/doi/abs/10.1111/ele.13788.

[38] Vaz, E., De Noronha, T. & Nijkamp, P. Exploratory Landscape Metrics for Agricultural Sustainability. *Agroecology and Sustainable Food Systems* **38**, 92–108 (2014). URL https://doi.org/10.1080/21683565.2013.825829.

[39] Yuan, Y. Modelling the spatial heterogeneity and molecular correlates of lymphocytic infiltration in triple-negative breast cancer. *Journal of The Royal Society Interface* **12**, 20141153 (2015). URL https://royalsocietypublishing.org/doi/10.1098/rsif.2014.1153.

[40] Yuan, Y. Spatial Heterogeneity in the Tumor Microenvironment. *Cold Spring Harbor Perspectives in Medicine* **6**, a026583 (2016). URL http://perspectivesinmedicine.cshlp.org/content/6/8/a026583.

[41] How Visual Search has transformed the modern shopping experience (2019). URL https://www.visenze.com/2019/03/21/how-visual-search-has-transformed-the-\modern-shopping-experience/.

[42] Bitirim, Y., Bitirim, S., Celik Ertugrul, D. & Toygar, O. An Evaluation of Reverse Image Search Performance of Google. In *2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC)*, 1368–1372 (2020).

[43] Hu, H. *et al.* Web-Scale Responsive Visual Search at Bing. *arXiv:1802.04914 [cs]* (2018). URL http://arxiv.org/abs/1802.04914.

[44] Hsiao, J.-H. & Li, L.-J. On visual similarity based interactive product recommendation for online shopping. In *2014 IEEE International Conference on Image Processing (ICIP)*, 3038–3041 (2014).

[45] Jing, Y. *et al.* Visual Search at Pinterest. *arXiv:1505.07647 [cs]* (2017). URL http://arxiv.org/abs/1505.07647.

[46] Bhardwaj, A. *et al.* Palette power: enabling visual search through colors. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '13, 1321–1329 (Association for Computing Machinery, New York, NY, USA, 2013). URL https://doi.org/10.1145/2487575.2488201.

[47] Yang, F. *et al.* Visual Search at eBay. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 2101–2110 (2017). URL http://arxiv.org/abs/1706.03154.

[48] Zhang, Y. *et al.* Visual Search at Alibaba. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* 993–1001 (2018). URL http://arxiv.org/abs/2102.04674.

[49] Fox, H. Is H&E morphology coming to an end? *Journal of Clinical Pathology* **53**, 38–40 (2000). URL https://jcp.bmj.com/content/53/1/38.

[50] Komura, D. & Ishikawa, S. Machine Learning Methods for Histopathological Image Analysis. *Computational and Structural Biotechnology Journal* **16**, 34–42 (2018). URL https://www.sciencedirect.com/science/article/pii/S2001037017300867.

[51] Madabhushi, A. & Lee, G. Image analysis and machine learning in digital pathology: Challenges and opportunities. *Medical Image Analysis* **33**, 170–175 (2016).

[52] Djuric, U., Zadeh, G., Aldape, K. & Diamandis, P. Precision histology: how deep learning is poised to revitalize histomorphology for personalized cancer care. *npj Precision Oncology* **1**, 1–5 (2017). URL https://www.nature.com/articles/s41698-017-0022-1.

[53] Xie, Y., Xing, F., Kong, X., Su, H. & Yang, L. Beyond Classification: Structured Regression for Robust Cell Detection Using Convolutional Neural Network. *Medical image computing and computer-assisted intervention : International Conference on Medical Image Computing and Computer-Assisted Intervention* **9351**, 358–365 (2015). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5226438/.

[54] Xu, J., Luo, X., Wang, G., Gilmore, H. & Madabhushi, A. A Deep Convolutional Neural Network for segmenting and classifying epithelial and stromal regions in histopathological images. *Neurocomputing* **191**, 214–223 (2016). URL https://www.sciencedirect.com/science/article/pii/S0925231216001004.

[55] Xu, J., Zhou, C., Lang, B. & Liu, Q. Deep Learning for Histopathological Image Analysis: Towards Computerized Diagnosis on Cancers. In Lu, L., Zheng, Y., Carneiro, G. & Yang, L. (eds.) *Deep Learning and Convolutional Neural Networks for Medical*

Image Computing: Precision Medicine, High Performance and Large-Scale Datasets, Advances in Computer Vision and Pattern Recognition, 73–95 (Springer International Publishing, Cham, 2017).

[56] Sharma, H., Zerbe, N., Klempert, I., Hellwich, O. & Hufnagl, P. Deep convolutional neural networks for automatic classification of gastric carcinoma using whole slide images in digital histopathology. *Computerized Medical Imaging and Graphics* **61**, 2–13 (2017). URL `https://www.sciencedirect.com/science/article/pii/S0895611117300502`.

[57] Qu, J. *et al.* Gastric Pathology Image Classification Using Stepwise Fine-Tuning for Deep Neural Networks. *Journal of Healthcare Engineering* **2018**, 8961781 (2018).

[58] Xu, Y. *et al.* Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC Bioinformatics* **18**, 281 (2017). URL `https://doi.org/10.1186/s12859-017-1685-x`.

[59] Xing, F., Xie, Y. & Yang, L. An Automatic Learning-Based Framework for Robust Nucleus Segmentation. *IEEE Transactions on Medical Imaging* **35**, 550–566 (2016).

[60] Sirinukunwattana, K. *et al.* Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images. *IEEE Transactions on Medical Imaging* **35**, 1196–1206 (2016).

[61] Pocevičiūtė, M., Eilertsen, G. & Lundström, C. Survey of XAI in digital pathology. *arXiv:2008.06353 [cs, eess]* **12090**, 56–88 (2020). URL `http://arxiv.org/abs/2008.06353`.

[62] Pocevičiūtė, M., Eilertsen, G., Jarkman, S. & Lundström, C. Can uncertainty boost the reliability of AI-based diagnostic methods in digital pathology? *arXiv:2112.09693 [cs, eess]* (2021). URL `http://arxiv.org/abs/2112.09693`.

[63] Thagaard, J. *et al.* Can You Trust Predictive Uncertainty Under Real Dataset Shifts in Digital Pathology? In Martel, A. L. *et al.* (eds.) *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, Lecture Notes in Computer Science, 824–833 (Springer International Publishing, Cham, 2020).

[64] Spanhol, F. A., Oliveira, L. S., Petitjean, C. & Heutte, L. A Dataset for Breast Cancer Histopathological Image Classification. *IEEE Transactions on Biomedical Engineering* **63**, 1455–1462 (2016).

[65] Kather, J. N. *et al.* Multi-class texture analysis in colorectal cancer histology. *Scientific Reports* **6**, 27988 (2016). URL `https://www.nature.com/articles/srep27988`.

[66] Han, Z. *et al.* Breast Cancer Multi-classification from Histopathological Images with Structured Deep Learning Model. *Scientific Reports* **7**, 4172 (2017). URL `https://www.nature.com/articles/s41598-017-04075-z`.

[67] Bayramoglu, N., Kannala, J. & Heikkilä, J. Deep learning for magnification independent breast cancer histopathology image classification. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2440–2445 (2016).

[68] Nalisnik, M. *et al.* Interactive phenotyping of large-scale histology imaging data with HistomicsML. *Scientific Reports* **7**, 14588 (2017). URL `https://www.nature.com/articles/s41598-017-15092-3`.

[69] Doyle, S., Monaco, J., Feldman, M., Tomaszewski, J. & Madabhushi, A. An active learning based classification strategy for the minority class problem: application to histopathology annotation. *BMC Bioinformatics* **12**, 424 (2011). URL `https://doi.org/10.1186/1471-2105-12-424`.

[70] Padmanabhan, R. K. *et al.* An Active Learning Approach for Rapid Characterization of Endothelial Cells in Human Tumors. *PLOS ONE* **9**, e90495 (2014). URL `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0090495`.

[71] Zhu, Y., Zhang, S., Liu, W. & Metaxas, D. N. Scalable histopathological image analysis via active learning. *Medical image computing and computer-assisted intervention: International Conference on Medical Image Computing and Computer-Assisted Intervention* **17**, 369–376 (2014).

[72] Xu, Y., Zhu, J.-Y., Chang, E. I.-C., Lai, M. & Tu, Z. Weakly supervised histopathology cancer image segmentation and classification. *Medical Image Analysis* **18**, 591–604 (2014). URL `https://www.sciencedirect.com/science/article/pii/S1361841514000188`.

[73] Shao, W., Sun, L. & Zhang, D. Deep active learning for nucleus classification in pathology images. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 199–202 (2018).

[74] Du, B., Qi, Q., Zheng, H., Huang, Y. & Ding, X. Breast Cancer Histopathological Image Classification via Deep Active Learning and Confidence Boosting. In Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L. & Maglogiannis, I. (eds.) *Artificial Neural Networks and Machine Learning – ICANN 2018*, Lecture Notes in Computer Science, 109–116 (Springer International Publishing, Cham, 2018).

[75] Smailagic, A. *et al.* MedAL: Deep Active Learning Sampling Method for Medical Image Analysis. *arXiv:1809.09287 [cs]* (2018). URL `http://arxiv.org/abs/1809.09287`.

[76] Hou, L. *et al.* Patch-based Convolutional Neural Network for Whole Slide Tissue Image Classification. *arXiv:1504.07947 [cs]* (2016). URL `http://arxiv.org/abs/1504.07947`.

[77] Karimi, D., Dou, H., Warfield, S. K. & Gholipour, A. Deep learning with noisy labels: exploring techniques and remedies in medical image analysis. *arXiv:1912.02911 [cs, eess, stat]* (2020). URL `http://arxiv.org/abs/1912.02911`.

[78] Le, H. *et al.* Pancreatic Cancer Detection in Whole Slide Images Using Noisy Label Annotations. In Shen, D. *et al.* (eds.) *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, Lecture Notes in Computer Science, 541–549 (Springer International Publishing, Cham, 2019).

[79] Tashk, A., Helfroush, M. S., Danyali, H. & Akbarzadeh, M. An automatic mitosis detection method for breast cancer histopathology slide images based on objective and pixel-wise textural features classification. In *The 5th Conference on Information and Knowledge Technology*, 406–410 (2013).

[80] Coudray, N. *et al.* Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nature Medicine* **24**, 1559–1567 (2018). URL `http://www.nature.com/articles/s41591-018-0177-5`.

[81] Korbar, B. *et al.* Deep Learning for Classification of Colorectal Polyps on Whole-slide Images. *Journal of Pathology Informatics* **8**, 30 (2017). URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5545773/`.

[82] Binnewies, M. *et al.* Understanding the tumor immune microenvironment (TIME) for effective therapy. *Nature medicine* **24**, 541–550 (2018). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5998822/.

[83] Griffiths, A. J. *et al. An Introduction to Genetic Analysis* (W. H. Freeman, 2000), 7th edn.

[84] Abedi, S. *et al.* Estimating the Survival of Patients With Lung Cancer: What Is the Best Statistical Model? *Journal of Preventive Medicine and Public Health* **52**, 140–144 (2019). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6459760/.

[85] Hassan, M. R. A. *et al.* Survival Analysis and Prognostic Factors for Colorectal Cancer Patients in Malaysia. *Asian Pacific journal of cancer prevention: APJCP* **17**, 3575–3581 (2016).

[86] Anderson, N. M. & Simon, M. C. The tumor microenvironment. *Current Biology* **30**, R921–R925 (2020). URL https://www.cell.com/current-biology/abstract/S0960-9822(20)30933-7.

[87] Shi, R., Tang, Y.-Q. & Miao, H. Metabolism in tumor microenvironment: Implications for cancer immunotherapy. *MedComm* **1**, 47–68 (2020). URL https://onlinelibrary.wiley.com/doi/abs/10.1002/mco2.6.

[88] Bade, B. C. & Dela Cruz, C. S. Lung Cancer 2020: Epidemiology, Etiology, and Prevention. *Clinics in Chest Medicine* **41**, 1–24 (2020). URL https://www.sciencedirect.com/science/article/pii/S0272523119300802.

[89] WHO Cancer Factsheet. URL https://www.who.int/news-room/fact-sheets/detail\/cancer.

[90] Ruiz-Cordero, R. & Devine, W. P. Targeted Therapy and Checkpoint Immunotherapy in Lung Cancer. *Surgical Pathology Clinics* **13**, 17–33 (2020). URL https://www.surgpath.theclinics.com/article/S1875-9181(19)30088-1/abstract.

[91] Imyanitov, E. N., Iyevleva, A. G. & Levchenko, E. V. Molecular testing and targeted therapy for non-small cell lung cancer: Current status and perspectives. *Critical Reviews in Oncology/Hematology* **157**, 103194 (2021). URL https://www.sciencedirect.com/science/article/pii/S1040842820303309.

[92] Altorki, N. K. *et al.* The lung microenvironment: an important regulator of tumour growth and metastasis. *Nature Reviews Cancer* **19**, 9–31 (2019). URL https://www.nature.com/articles/s41568-018-0081-9.

[93] Zhang, C., Liu, Y., Guo, S. & Zhang, J. Different biomarkers in non-small cell lung cancer in blood vessel invasion. *Cell Biochemistry and Biophysics* **70**, 777–784 (2014).

[94] Chan, T. A. *et al.* Development of tumor mutation burden as an immunotherapy biomarker: utility for the oncology clinic. *Annals of Oncology* **30**, 44–56 (2019). URL https://www.annalsofoncology.org/article/S0923-7534(19)30997-4/abstract.

[95] Rehman, J. A. *et al.* Quantitative and pathologist-read comparison of the heterogeneity of programmed death-ligand 1 (PD-L1) expression in non-small cell lung cancer. *Modern Pathology* **30**, 340–349 (2017). URL https://www.nature.com/articles/modpathol2016186.

[96] Huang, Y.-K. *et al.* Macrophage spatial heterogeneity in gastric cancer defined by multiplex immunohistochemistry. *Nature Communications* **10**, 3928 (2019). URL https://www.nature.com/articles/s41467-019-11788-4.

[97] Heindl, A. *et al.* Relevance of Spatial Heterogeneity of Immune Infiltration for Predicting Risk of Recurrence After Endocrine Therapy of ER+ Breast Cancer. *JNCI: Journal of the National Cancer Institute* **110**, 166–175 (2018). URL https://doi.org/10.1093/jnci/djx137.

[98] Rudolf, J. *et al.* Regulatory T cells and cytotoxic T cells close to the epithelial–stromal interface are associated with a favorable prognosis. *OncoImmunology* **9**, 1746149 (2020). URL https://doi.org/10.1080/2162402X.2020.1746149.

[99] Valous, N. A., Moraleda, R. R., Jäger, D., Zörnig, I. & Halama, N. Interrogating the microenvironmental landscape of tumors with computational image analysis approaches. *Seminars in Immunology* **48**, 101411 (2020). URL https://linkinghub.elsevier.com/retrieve/pii/S1044532320300270.

[100] Saltz, J. Spatial Organization and Molecular Correlation of Tumor-Infiltrating Lymphocytes Using Deep Learning on Pathology Images 21 (2018).

[101] Heindl, A., Lan, C., Rodrigues, D. N., Koelble, K. & Yuan, Y. Similarity and diversity of the tumor microenvironment in multiple metastases: critical implications for overall and progression-free survival of high-grade serous ovarian cancer. *Oncotarget* **7**, 71123–71135 (2016).

[102] Corredor, G. *et al.* Spatial Architecture and Arrangement of Tumor-Infiltrating Lymphocytes for Predicting Likelihood of Recurrence in Early-Stage Non–Small Cell Lung Cancer. *Clinical Cancer Research* **25**, 1526–1534 (2019).

[103] Xi, K.-X. *et al.* Tumor-stroma ratio (TSR) in non-small cell lung cancer (NSCLC) patients after lung resection is a prognostic factor for survival. *Journal of Thoracic Disease* **9**, 4017–4026 (2017). URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5723873/`.

[104] Mobadersany, P. *et al.* Predicting cancer outcomes from histology and genomics using convolutional networks. *MEDICAL SCIENCES* 10 (2018).

[105] Yousefi, S. *et al.* Predicting clinical outcomes from large scale cancer genomic profiles with deep survival models. *Scientific Reports* **7**, 11707 (2017). URL `https://www.nature.com/articles/s41598-017-11817-6`.

[106] Chen, M. *et al.* Classification and mutation prediction based on histopathology H&E images in liver cancer using deep learning. *npj Precision Oncology* **4**, 1–7 (2020). URL `https://www.nature.com/articles/s41698-020-0120-3`.

[107] Liao, H. *et al.* Deep learning-based classification and mutation prediction from histopathological images of hepatocellular carcinoma. *Clinical and Translational Medicine* **10** (2020). URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7403820/`.

[108] Chen, X., Lin, X., Shen, Q. & Qian, X. Combined Spiral Transformation and Model-Driven Multi-Modal Deep Learning Scheme for Automatic Prediction of TP53 Mutation in Pancreatic Cancer. *IEEE Transactions on Medical Imaging* **40**, 735–747 (2021).

[109] Wu, Z., Huang, X., Huang, S., Ding, X. & Wang, L. Direct Prediction of BRAFV600E Mutation from Histopathological Images in Papillary Thyroid Carcinoma with a Deep Learning Workflow. In *2020 4th International Conference on Computer Science and*

Artificial Intelligence, CSAI 2020, 146–151 (Association for Computing Machinery, New York, NY, USA, 2020). URL https://doi.org/10.1145/3445815.3445840.

[110] Schaumberg, A. J., Rubin, M. A. & Fuchs, T. J. H&E-stained Whole Slide Image Deep Learning Predicts SPOP Mutation State in Prostate Cancer. preprint, Pathology (2016). URL http://biorxiv.org/lookup/doi/10.1101/064279.

[111] Kather, J. N. *et al.* Pan-cancer image-based detection of clinically actionable genetic alterations. *Nature Cancer* **1**, 789–799 (2020). URL http://www.nature.com/articles/s43018-020-0087-6.

[112] Diao, J. A. *et al.* Human-interpretable image features derived from densely mapped cancer pathology slides predict diverse molecular phenotypes. *Nature Communications* **12**, 1613 (2021). URL http://www.nature.com/articles/s41467-021-21896-9.

[113] Lafferty, J. D., McCallum, A. & Pereira, F. C. N. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, 282–289 (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001).

[114] Manivannan, S. *et al.* Brain Tumor Region Segmentation using Local Co-occurrence Features and Conditional Random Fields 5 (2014).

[115] Wang, J., MacKenzie, J. D., Ramachandran, R. & Chen, D. Z. A Deep Learning Approach for Semantic Segmentation in Histology Tissue Images. In Ourselin, S., Joskowicz, L., Sabuncu, M. R., Unal, G. & Wells, W. (eds.) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, Lecture Notes in Computer Science, 176–184 (Springer International Publishing, Cham, 2016).

[116] Liang, Q. *et al.* Weakly Supervised Biomedical Image Segmentation by Reiterative Learning. *IEEE Journal of Biomedical and Health Informatics* **23**, 1205–1214 (2019).

[117] Zormpas-Petridis, K. *et al.* Superpixel-Based Conditional Random Fields (SuperCRF): Incorporating Global and Local Context for Enhanced Deep Learning in Melanoma Histopathology. *Frontiers in Oncology* **9**, 1045 (2019). URL https://www.frontiersin.org/article/10.3389/fonc.2019.01045.

[118] Li, Y. *et al.* Automated Gleason Grading and Gleason Pattern Region Segmentation Based on Deep Learning for Pathological Images of Prostate Cancer. *IEEE Access* **8**, 117714–117725 (2020).

[119] Li, Y. & Ping, W. Cancer Metastasis Detection With Neural Conditional Random Field. *arXiv:1806.07064 [cs]* (2018). URL `http://arxiv.org/abs/1806.07064`.

[120] Qu, H. *et al.* Weakly Supervised Deep Nuclei Segmentation using Points Annotation in Histopathology Images. In *Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning*, 390–400 (PMLR, 2019). URL `https://proceedings.mlr.press/v102/qu19a.html`.

[121] Dong, J., Guo, X. & Wang, G. GECNN-CRF for Prostate Cancer Detection with WSI. In Jia, Y., Zhang, W. & Fu, Y. (eds.) *Proceedings of 2020 Chinese Intelligent Systems Conference*, Lecture Notes in Electrical Engineering, 646–658 (Springer, Singapore, 2021).

[122] Wróblewska, A. & Rączkowski, Ł. Visual Recommendation Use Case for an Online Marketplace Platform: allegro.pl. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, SIGIR '16, 591–594 (Association for Computing Machinery, New York, NY, USA, 2016). URL `https://doi.org/10.1145/2911451.2926722`.

[123] Rączkowski, Ł., Możejko, M., Zambonelli, J. & Szczurek, E. ARA: accurate, reliable and active histopathological image classification framework with Bayesian deep learning. *Scientific Reports* **9**, 14347 (2019). URL `https://www.nature.com/articles/s41598-019-50587-1`.

[124] Rączkowski, Ł. *et al.* Deep learning-based tumor microenvironment segmentation is predictive of tumor mutations and patient survival in non-small-cell lung cancer. Tech. Rep. (2021). URL `https://www.biorxiv.org/content/10.1101/2021.10.09.462574v1`.

[125] Lähnemann, D. *et al.* Eleven grand challenges in single-cell data science. *Genome Biology* **21**, 31 (2020). URL `https://doi.org/10.1186/s13059-020-1926-6`.

[126] Huang, J., Kumar, S., Mitra, M., Zhu, W.-J. & Zabih, R. Image indexing using color correlograms. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 762–768 (1997).

[127] Won, C. S., Park, D. K. & Park, S.-J. Efficient Use of MPEG-7 Edge Histogram Descriptor. *ETRI Journal* **24**, 23–30 (2002). URL https://onlinelibrary.wiley.com/doi/abs/10.4218/etrij.02.0102.0103.

[128] Bosch, A., Zisserman, A. & Munoz, X. Representing shape with a spatial pyramid kernel. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, CIVR '07, 401–408 (Association for Computing Machinery, New York, NY, USA, 2007). URL https://doi.org/10.1145/1282280.1282340.

[129] Lux, M. Content based image retrieval with LIRe. In *Proceedings of the 19th ACM international conference on Multimedia*, MM '11, 735–738 (Association for Computing Machinery, New York, NY, USA, 2011). URL https://doi.org/10.1145/2072298.2072432.

[130] Canny, J. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-8**, 679–698 (1986).

[131] Ojala, T., Pietikainen, M. & Harwood, D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In *Proceedings of 12th International Conference on Pattern Recognition*, vol. 1, 582–585 vol.1 (1994).

[132] Ojala, T., Pietikäinen, M. & Harwood, D. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition* **29**, 51–59 (1996). URL https://www.sciencedirect.com/science/article/pii/0031320395000674.

[133] Rogers, D. J. & Tanimoto, T. T. A Computer Program for Classifying Plants. *Science* **132**, 1115–1118 (1960). URL https://www.science.org/lookup/doi/10.1126/science.132.3434.1115.

[134] Rubner, Y., Tomasi, C. & Guibas, L. A metric for distributions with applications to image databases. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, 59–66 (1998).

[135] Hazewinkel, M. *Encyclopaedia of mathematics* (Springer-Verlag, Berlin; New York, 2002). URL http://eom.springer.de/default.htm.

[136] Elasticsearch. URL https://github.com/elastic/elasticsearch.

[137] OpenCV. URL https://github.com/opencv/opencv.

[138] Wang, K. Image Plugin for Elasticsearch. URL https://github.com/kzwang/elasticsearch-image.

[139] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015). URL http://www.nature.com/articles/nature14539.

[140] Chollet, F. *et al.* Keras. https://keras.io (2015).

[141] Zhou & Chellappa. Computation of optical flow using a neural network. In *IEEE 1988 International Conference on Neural Networks*, 71–78 vol.2 (1988).

[142] Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (The MIT Press, 2016).

[143] Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference* **90**, 227–244 (2000). URL https://www.sciencedirect.com/science/article/pii/S0378375800001154.

[144] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986). URL https://www.nature.com/articles/323533a0.

[145] Rumelhart, D. E., McClelland, J. L. & Group, P. R. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, vol. 1 (A Bradford Book, Cambridge, MA, USA, 1986).

[146] He, K. & Sun, J. Convolutional Neural Networks at Constrained Time Cost (2014). URL https://arxiv.org/abs/1412.1710v1.

[147] Srivastava, R. K., Greff, K. & Schmidhuber, J. Highway Networks (2015). URL https://arxiv.org/abs/1505.00387v2.

[148] Thomson, A. Neocortical layer 6, a review. *Frontiers in Neuroanatomy* **4**, 13 (2010). URL https://www.frontiersin.org/article/10.3389/fnana.2010.00013.

[149] Robbins, H. & Monro, S. A Stochastic Approximation Method. *The Annals of Mathematical Statistics* **22**, 400–407 (1951). URL https://projecteuclid.org/journals/annals-of-mathematical-statistics\/volume-22/issue-3/A-Stochastic-Approximation-Method/10.1214/aoms\/1177729586.full.

[150] Kiefer, J. & Wolfowitz, J. Stochastic Estimation of the Maximum of a Regression Function. *The Annals of Mathematical Statistics* **23**, 462–466 (1952). URL https://projecteuclid.org/journals/annals-of-mathematical-statistics\/volume-23/issue-3/Stochastic-Estimation-of-the-Maximum-of-a-Regression-\Function/10.1214/aoms/1177729392.full.

[151] Duchi, J., Hazan, E. & Singer, Y. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research* **12**, 2121–2159 (2011). URL http://jmlr.org/papers/v12/duchi11a.html.

[152] Tieleman, T. & Hinton, G. Lecture 6.5 - RMSProp, COURSERA: Neural Networks for Machine Learning (2012).

[153] Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]* (2017). URL http://arxiv.org/abs/1412.6980.

[154] Ruder, S. An overview of gradient descent optimization algorithms (2016). URL https://arxiv.org/abs/1609.04747v2.

[155] Gal, Y. & Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *arXiv:1506.02142 [cs, stat]* (2016). URL http://arxiv.org/abs/1506.02142.

[156] Smith, L. & Gal, Y. Understanding Measures of Uncertainty for Adversarial Example Detection. *arXiv:1803.08533 [cs, stat]* (2018). URL http://arxiv.org/abs/1803.08533.

[157] Wittekind, D. Traditional staining for routine diagnostic pathology including the role of tannic acid. 1. Value and limitations of the hematoxylin-eosin stain. *Biotechnic & Histochemistry* **78**, 261–270 (2003). URL http://www.tandfonline.com/doi/full/10.1080/10520290310001633725.

[158] Titford, M. Progress in the Development of Microscopical Techniques for Diagnostic Pathology. *Journal of Histotechnology* **32**, 9–19 (2009). URL https://doi.org/10.1179/his.2009.32.1.9.

[159] Xu, B., Wang, N., Chen, T. & Li, M. Empirical Evaluation of Rectified Activations in Convolutional Network. *arXiv:1505.00853 [cs, stat]* (2015). URL http://arxiv.org/abs/1505.00853.

[160] Ringach, D. & Shapley, R. Reverse correlation in neurophysiology. *Cognitive Science* **28**, 147–166 (2004). URL https://www.sciencedirect.com/science/article/pii/S0364021303001174.

[161] Olshausen, B. A. & Field, D. J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature* **381**, 607–609 (1996). URL https://www.nature.com/articles/381607a0.

[162] Ribeiro, M. G. *et al.* Classification of colorectal cancer based on the association of multidimensional and multiresolution features. *Expert Systems with Applications* **120**, 262–278 (2019). URL https://www.sciencedirect.com/science/article/pii/S0957417418307541.

[163] Wang, C., Shi, J., Zhang, Q. & Ying, S. Histopathological image classification with bilinear convolutional neural networks. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 4050–4053 (2017).

[164] Pham, T. D. Scaling of Texture in Training Autoencoders for Classification of Histological Images of Colorectal Cancer. In Cong, F., Leung, A. & Wei, Q. (eds.) *Advances in Neural Networks - ISNN 2017*, Lecture Notes in Computer Science, 524–532 (Springer International Publishing, Cham, 2017).

[165] Ciompi, F. *et al.* The importance of stain normalization in colorectal tissue classification with convolutional networks. *arXiv:1702.05931 [cs]* (2017). URL http://arxiv.org/abs/1702.05931.

[166] Goel, P. & Chen, L. On the Robustness of Monte Carlo Dropout Trained with Noisy Labels. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2219–2228 (IEEE, Nashville, TN, USA, 2021). URL https://ieeexplore.ieee.org/document/9523060/.

[167] Hanahan, D. & Weinberg, R. A. The Hallmarks of Cancer. *Cell* **100**, 57–70 (2000). URL https://www.cell.com/cell/abstract/S0092-8674(00)81683-9.

[168] Brown, T. A. *Genomes 4* (Garland Science, New York, NY, 2017), 4th edition edn.

[169] Hoadley, K. A. *et al.* Cell-of-Origin Patterns Dominate the Molecular Classification of 10,000 Tumors from 33 Types of Cancer. *Cell* **173**, 291–304.e6 (2018). URL https://www.cell.com/cell/abstract/S0092-8674(18)30302-7.

[170] McLendon, R. *et al.* Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature* **455**, 1061–1068 (2008). URL https://www.nature.com/articles/nature07385.

[171] Bell, D. *et al.* Integrated genomic analyses of ovarian carcinoma. *Nature* **474**, 609–615 (2011). URL https://www.nature.com/articles/nature10166.

[172] Koboldt, D. C. *et al.* Comprehensive molecular portraits of human breast tumours. *Nature* **490**, 61–70 (2012). URL https://www.nature.com/articles/nature11412.

[173] Biankin, A. V. *et al.* Pancreatic cancer genomes reveal aberrations in axon guidance pathway genes. *Nature* **491**, 399–405 (2012). URL https://www.nature.com/articles/nature11547.

[174] Duma, N., Santana-Davila, R. & Molina, J. R. Non–Small Cell Lung Cancer: Epidemiology, Screening, Diagnosis, and Treatment. *Mayo Clinic Proceedings* **94**, 1623–1640 (2019). URL https://www.mayoclinicproceedings.org/article/S0025-6196(19)30070-9\/abstract.

[175] Travis, W. D., Brambilla, E., Burke, A. P., Marx, A. & Nicholson, A. G. Introduction to The 2015 World Health Organization Classification of Tumors of the Lung, Pleura, Thymus, and Heart. *Journal of Thoracic Oncology: Official Publication of the International Association for the Study of Lung Cancer* **10**, 1240–1242 (2015).

[176] Bankhead, P. *et al.* QuPath: Open source software for digital pathology image analysis. *Scientific Reports* **7**, 16878 (2017). URL https://www.nature.com/articles/s41598-017-17204-5.

[177] Ramos, M. curatedTCGAData: Curated Data From The Cancer Genome Atlas (TCGA) as MultiAssayExperiment Objects (2020). URL https://bioconductor.org/packages/curatedTCGAData.

[178] Chen, J. *et al.* Genomic landscape of lung adenocarcinoma in East Asians. *Nature Genetics* **52**, 177–186 (2020). URL https://www.nature.com/articles/s41588-019-0569-6.

[179] Zhou, F. & Zhou, C. Lung cancer in never smokers—the East Asian experience. *Translational Lung Cancer Research* **7**, 450–463 (2018). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6131183/.

[180] Goldman, M. J. *et al.* Visualizing and interpreting cancer genomics data via the Xena platform. *Nature Biotechnology* **38**, 675–678 (2020). URL https://www.nature.com/articles/s41587-020-0546-8.

[181] SureSelect Cancer All-In-One Catalog and Custom Assays. URL https://www.agilent.com/en/product/next-generation-sequencing\/hybridization-based-next-generation-sequencing-ngs/cancer-all-in-one\-assays/sureselect-cancer-all-in-one-lung-assay-520074.

[182] Reinhard, E., Adhikhmin, M., Gooch, B. & Shirley, P. Color transfer between images. *IEEE Computer Graphics and Applications* **21**, 34–41 (2001).

[183] Macenko, M. *et al.* A method for normalizing histology slides for quantitative analysis. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 1107–1110 (2009).

[184] Vahadane, A. *et al.* Structure-Preserving Color Normalization and Sparse Stain Separation for Histological Images. *IEEE Transactions on Medical Imaging* **35**, 1962–1971 (2016).

[185] Simpson, E. H. Measurement of Diversity. *Nature* **163**, 688–688 (1949). URL https://www.nature.com/articles/163688a0.

[186] Shannon, C. E. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review* **5**, 3–55 (2001). URL https://doi.org/10.1145/584091.584093.

[187] Cox, D. R. Regression Models and Life-Tables. *Journal of the Royal Statistical Society. Series B (Methodological)* **34**, 187–220 (1972). URL https://www.jstor.org/stable/2985181.

[188] Harrell, F. E., Jr, Califf, R. M., Pryor, D. B., Lee, K. L. & Rosati, R. A. Evaluating the Yield of Medical Tests. *JAMA* **247**, 2543–2546 (1982). URL https://doi.org/10.1001/jama.1982.03320430047030.

[189] Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R. & Lin, C.-J. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research* **9**, 1871–1874 (2008). URL http://jmlr.org/papers/v9/fan08a.html.

[190] Breiman, L. Random Forests. *Machine Learning* **45**, 5–32 (2001). URL https://doi.org/10.1023/A:1010933404324.

[191] Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011). URL http://jmlr.org/papers/v12/pedregosa11a.html.

[192] Koller, D. & Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive Computation and Machine Learning series (MIT Press, Cambridge, MA, USA, 2009).

[193] Liu, T., Huang, X. & Ma, J. Conditional Random Fields for Image Labeling. *Mathematical Problems in Engineering* **2016**, e3846125 (2016). URL https://www.hindawi.com/journals/mpe/2016/3846125/.

[194] Ratliff, N. D., Bagnell, J. A. & Zinkevich, M. A. (approximate) subgradient methods for structured prediction. In Meila, M. & Shen, X. (eds.) *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, vol. 2 of *Proceedings of Machine Learning Research*, 380–387 (PMLR, San Juan, Puerto Rico, 2007). URL https://proceedings.mlr.press/v2/ratliff07a.html.

[195] Müller, A. C. & Behnke, S. pystruct - Learning Structured Prediction in Python. *Journal of Machine Learning Research* **15**, 2055–2060 (2014). URL http://jmlr.org/papers/v15/mueller14a.html.

[196] Martins, A. F. T., Figueiredo, M. A. T., Aguiar, P. M. Q., Smith, N. A. & Xing, E. P. Alternating Directions Dual Decomposition. *arXiv:1212.6550 [cs]* (2012). URL http://arxiv.org/abs/1212.6550.

[197] Shalev-Shwartz, S., Singer, Y., Srebro, N. & Cotter, A. Pegasos: primal estimated sub-gradient solver for SVM. *Mathematical Programming* **127**, 3–30 (2011). URL https://doi.org/10.1007/s10107-010-0420-4.

[198] Oliver, A. J. *et al.* Tissue-Dependent Tumor Microenvironments and Their Impact on Immunotherapy Responses. *Frontiers in Immunology* **9**, 70 (2018). URL http://journal.frontiersin.org/article/10.3389/fimmu.2018.00070/full.

[199] Mo, J. *et al.* Smokers or non-smokers: who benefits more from immune checkpoint inhibitors in treatment of malignancies? An up-to-date meta-analysis. *World Journal of Surgical Oncology* **18**, 15 (2020). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6971889/.

[200] Bian, C. *et al.* ImmunoAIzer: A Deep Learning-Based Computational Framework to Characterize Cell Distribution and Gene Mutation in Tumor Microenvironment. *Cancers* **13**, 1659 (2021). URL https://www.mdpi.com/2072-6694/13/7/1659.

[201] Courtiol, P. *et al.* Deep learning-based classification of mesothelioma improves prediction of patient outcome. *Nature Medicine* **25**, 1519–1525 (2019). URL https://www.nature.com/articles/s41591-019-0583-3.

[202] Wang, S., Lokhande, V., Singh, M., Kording, K. & Yarkony, J. End-to-end Training of CNN-CRF via Differentiable Dual-Decomposition. *arXiv:1912.02937 [cs]* (2019). URL http://arxiv.org/abs/1912.02937.

[203] Ma, X. & Hovy, E. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *arXiv:1603.01354 [cs, stat]* (2016). URL http://arxiv.org/abs/1603.01354.

[204] Colovic, A., Knöbelreiter, P., Shekhovtsov, A. & Pock, T. End-to-End Training of Hybrid CNN-CRF Models for Semantic Segmentation using Structured Learning: Computer Vision Winter Workshop (2017).

[205] Dosovitskiy, A. *et al.* An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929 [cs]* (2021). URL http://arxiv.org/abs/2010.11929.

[206] Kessler, R. *et al.* Blood vessel invasion is a major prognostic factor in resected non-small cell lung cancer. *The Annals of Thoracic Surgery* **62**, 1489–1493 (1996). URL https://linkinghub.elsevier.com/retrieve/pii/0003497596005401.

[207] Park, S. Y. *et al.* Tumor Necrosis as a Prognostic Factor for Stage IA Non-Small Cell Lung Cancer. *The Annals of Thoracic Surgery* **91**, 1668–1673 (2011). URL https://www.annalsthoracicsurgery.org/article/S0003-4975(10)02915-2\/abstract.

[208] Chen, Z., Fillmore, C. M., Hammerman, P. S., Kim, C. F. & Wong, K.-K. Non-small-cell lung cancers: a heterogeneous set of diseases. *Nature Reviews Cancer* **14**, 535–546 (2014). URL https://www.nature.com/articles/nrc3775.

[209] Nishino, M. *et al.* Histologic and cytomorphologic features of ALK-rearranged lung adenocarcinomas. *Modern Pathology: An Official Journal of the United States and Canadian Academy of Pathology, Inc* **25**, 1462–1472 (2012).

[210] Sholl, L. M. *et al.* ROS1 immunohistochemistry for detection of ROS1-rearranged lung adenocarcinomas. *The American Journal of Surgical Pathology* **37**, 1441–1449 (2013).

[211] Wang, R. *et al.* RET fusions define a unique molecular and clinicopathologic subtype of non-small-cell lung cancer. *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology* **30**, 4352–4359 (2012).

[212] Skoulidis, F. *et al.* STK11/LKB1 Mutations and PD-1 Inhibitor Resistance in KRAS-Mutant Lung Adenocarcinoma. *Cancer Discovery* **8**, 822–835 (2018).

[213] Papillon-Cavanagh, S., Doshi, P., Dobrin, R., Szustakowski, J. & Walsh, A. M. STK11 and KEAP1 mutations as prognostic biomarkers in an observational real-world lung adenocarcinoma cohort. *ESMO open* **5**, e000706 (2020).

[214] Błach, J., Wojas-Krawczyk, K., Nicoś, M. & Krawczyk, P. Failure of Immunotherapy—The Molecular and Immunological Origin of Immunotherapy Resistance in Lung Cancer. *International Journal of Molecular Sciences* **22**, 9030 (2021). URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8396490/.

[215] Drusbosky, L. M., Rodriguez, E., Dawar, R. & Ikpeazu, C. V. Therapeutic strategies in RET gene rearranged non-small cell lung cancer. *Journal of Hematology & Oncology* **14**, 50 (2021). URL https://doi.org/10.1186/s13045-021-01063-9.

[216] Drilon, A. *et al.* Frequency of Brain Metastases and Multikinase Inhibitor Outcomes in Patients With RET–Rearranged Lung Cancers. *Journal of Thoracic Oncology* **13**,

1595–1601 (2018). URL https://www.jto.org/article/S1556-0864(18)30780-9/abstract.

[217] Paulsson, J., Ehnman, M. & Östman, A. PDGF receptors in tumor biology: prognostic and predictive potential. *Future Oncology (London, England)* **10**, 1695–1708 (2014).

[218] Kilvaer, T. K. *et al.* Differential prognostic impact of platelet-derived growth factor receptor expression in NSCLC. *Scientific Reports* **9**, 10163 (2019). URL https://www.nature.com/articles/s41598-019-46510-3.

[219] Greenwald, N. F. *et al.* Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature Biotechnology* 1–11 (2021). URL https://www.nature.com/articles/s41587-021-01094-0.

[220] Jackson, H. W. *et al.* The single-cell pathology landscape of breast cancer. *Nature* **578**, 615–620 (2020). URL https://www.nature.com/articles/s41586-019-1876-x.

[221] Martin-Gonzalez, P., Crispin-Ortuzar, M. & Markowetz, F. Predictive Modelling of Highly Multiplexed Tumour Tissue Images by Graph Neural Networks. In Reyes, M. *et al.* (eds.) *Interpretability of Machine Intelligence in Medical Image Computing, and Topological Data Analysis and Its Applications for Medical Data*, Lecture Notes in Computer Science, 98–107 (Springer International Publishing, Cham, 2021).

[222] Schürch, C. M. *et al.* Coordinated Cellular Neighborhoods Orchestrate Antitumoral Immunity at the Colorectal Cancer Invasive Front. *Cell* **182**, 1341–1359.e19 (2020). URL https://www.cell.com/cell/abstract/S0092-8674(20)30870-9.