Neural image compression for non-small cell lung cancer subtype classification in H&E stained whole-slide images

Witali Aswolinskiy*, David Tellez, Gabriel Raya, Lieke van der Woude, Monika Looijen-Salamon, Jeroen van der Laak, Katrien Grunberg, Francesco Ciompi

Computational Pathology Group, Radboud University Medical Center, The Netherlands
*witali.aswolinskiy@radboudumc.nl

ABSTRACT

Classification of non-small-cell lung cancer (NSCLC) into adenocarcinoma (LUAD) and squamous cell carcinoma (LUSC) via histopathology is a vital prerequisite to select the appropriate treatment for lung cancer patients. Most machine learning approaches rely on manually annotating large numbers of whole slide images (WSI) for training. However, manually delineating cancer areas or even single cancer cells on hundreds or thousands of slides is tedious, subjective and requires highly trained pathologists. We propose to use Neural Image Compression (NIC), which requires only slide-level labels, to classify NSCLC into LUSC and LUAD. NIC consists of two phases/networks. In the first phase the slides are compressed with a convolutional neural network (CNN) acting as an *encoder*. In the second phase the compressed slides are classified with a second CNN. We trained our classification model on >2,000 NIC-compressed slides from the TCGA and TCIA databases and evaluated the model performance additionally on several internal and external cohorts. We show that NIC approaches state of the art performance on lung cancer classification, with an average AUC of 0.94 on the TCGA and TCIA testdata, and AUCs between 0.84 and 0.98 on other independent datasets.

Keywords: Computational Pathology, Lung cancer, Deep learning, Neural Image Compression

1. INTRODUCTION

Lung cancer is the third most common cancer in men and women and the leading cause of cancer-related deaths worldwide.[1] About 80-85% of all lung cancers is non-small cell lung cancer (NSCLC).[2] The main subtypes of NSCLC are adenocarcinoma (LUAD), squamous cell carcinoma (LUSC) and large cell carcinoma, with LUAD and LUSC representing the vast majority of lung cancer cases. The subtype identification is important to determine the best treatment options for the patient. Here, we focus on classification of LUAD versus LUSC.

Deep Learning on whole slide images (WSI) can automate or augment the histological analysis and reduce its time, effort and subjective bias. Most deep learning approaches, however, require experienced pathologists to create many pixel- or patch-level annotations of the tissue type, which is very tedious and time-consuming.[3] The alternative is a weakly supervised learning approach to image classification, where only slide-level labels are used to train the deep learning model. In the context of weakly supervised learning, some approaches are based on the simplifying assumption that all patches in a slide are predictive for the slide-level label.[4,5] After training a network to predict the slide label for each patch of the slide, the patch predictions are aggregated to form a prediction for the whole slide. These approaches have been successfully applied to lung cancer classification[4] including mutation prediction.[5] A more nuanced approach is pursued in Multiple Instance Learning (MIL). In MIL, a slide is viewed as a collection of small patches, where only a few patches or even a single one determine the slide label. A MIL-approach developed to separate normal from cancerous slides was successfully evaluated on a large-scale datasets of prostate cancer, basal cell carcinoma and breast cancer metastases.[6] This approach was recently extended to multi-class classification with attention (CLAM[7]), achieving a high accuracy in lung and kidney cancer subtyping. The main components of CLAM are attention-weighted pooling and attention-driven clustering.

While effective in many scenarios, MIL-based approaches ignore the possibility of global slide-level patterns being important for prediction, since the slide-level classifier sees only tiny fractions (i.e., patches) of the slide at a time. Giving a neural network the opportunity to 'see' the complete slide is however challenging due to the

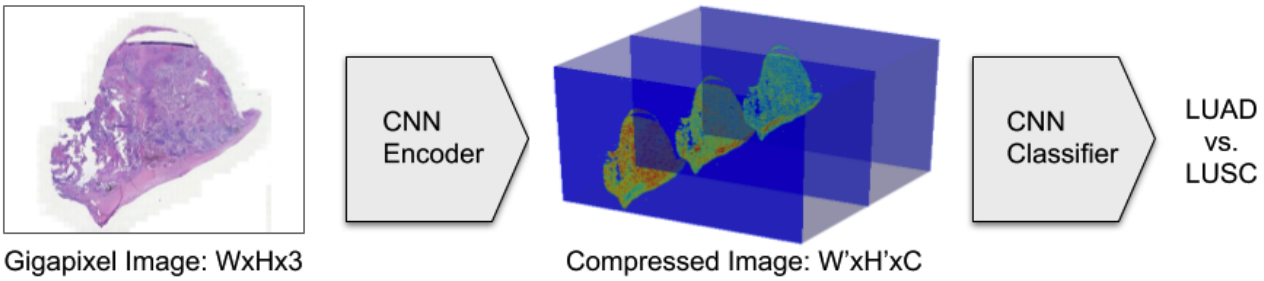


Figure 1: Overview of the NIC framework. The CNN Encoder compresses the whole slide image in the spatial dimensions W and H to W'≪W, H'≪H while increasing the depth to C. The CNN Classifier classifies the compressed image.

large slide sizes. Naively, it would require several hundred Gigabytes of RAM to process slides completely. Neural Image Compression (NIC)[8] bypasses this problem by strongly compressing slides before processing them.

Here, we propose to use NIC to classify whole slide images into LUSC and LUAD. We show that it can approach state-of-the-art performance currently set by CLAM[7] on publicly available datasets of lung cancer histopathology images from two different sources, namely the TCGA and the TCIA archives. For comparison, we set up the training and evaluation routine similar to the CLAM experiments for lung subtype classification as well as test our method on additional independent datasets.

2. MATERIALS

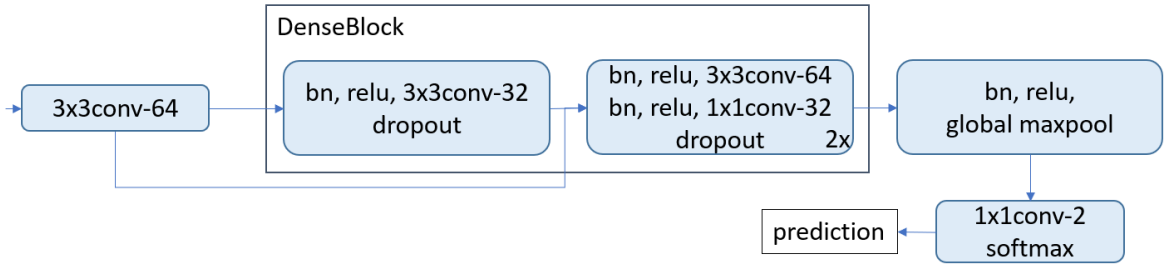
For model training, we collected 2409 NSCLC resection slides from The Cancer Imaging Archive (TCIA)[9] and The Cancer Genome Atlas (TCGA).[10] The slides from TCIA were part of a program by the National Cancer Institute Clinical Proteomic Tumor Analysis Consortium (CPTAC). The combined TCGA + CPTAC datasets were used for model training and testing, as done in.[7] For further testing, we included additional independent datasets from multiple sources: a) n=64 resection slides from the computational precision medicine challenge at the MICCAI 2017 conference;[11] b) n=60 NSCLC resection slides from Radboud University Medical Center, Nijmegen (Netherlands), which we name here "Radboud60"; c) n=103 of the 150 slides from the training set of the ACDC challenge.[12] These slides were provided with manual annotations of tumor regions, which were not used in this project. Since no information about the lung cancer subtype was provided in ACDC, a lung pathologist (KG) checked all ACDC training slides and determined the subtype. Among those, 103 slides were found to be either LUAD or LUSC, which were finally used in this study. Furthermore, we divided those slides into resections (32) and biopsies (71). Table 1 provides an overview of the slide numbers and the split in training, validation and testing, where appropriate (only data from TCGA and CPTAC was used for training). To the best of our knowledge, there is no overlap between the datasets. Similar to the CLAM experiments, we split the TCGA and CPTAC data into traninig, validation and testing randomly ten times and trained for each split a model independently.

Dataset	LUSC	LUAD	Training/Validation/Testing
TCGA	505	531	80/10/10 (%)
CPTAC	689	684	80/10/10 (%)
MICCAI	32	32	-/-/100 (%)
ACDC resections	15	17	-/-/100 (%)
ACDC biopsies	49	22	-/-/100 (%)
Radboud60	28	32	-/-/100 (%)

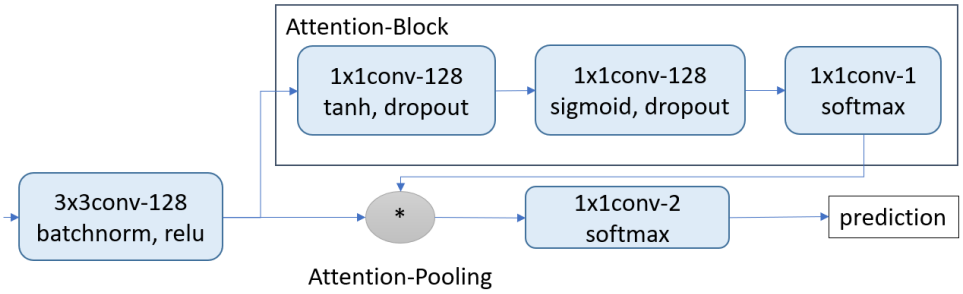
Table 1: Overview of the datasets used in this study.

3. METHODS

A whole slide image can be as large as 50.000x50.000 pixels or more and is often referred to as a gigapixel image. Since it is not possible to fit the entire slide into memory on modern GPU hardware, most deep learning approaches work either on smaller sub-crops or patches or try to reduce the input size. The latter is the main



(a) NIC-D



(b) NIC-A

Figure 2: Evaluated NIC-classifier architectures: NIC-D (top) and NIC-A (bottom); the shorthand notation fx$f$conv-n stands for a fxf convolution with n output channels. The attention-pooling operation ($*$) stands for the sum of all patch-feature-vectors weighted by the attention-scores (one attention weight per patch-feature).

focus of Neural Image Compression[8,13] for WSI classification (see Fig. 1). It consists of two phases: First, the slides are compressed using a pretrained convolutional neural network. This encoder is responsible for extracting useful features as well as accounting for stain variation. Then, a second network is trained on the compressed slides to predict their labels. In our experiments, we compressed a 128x128x3 patch to 128 features, achieving a compression factor of 384, e.g. reducing a slide of 50,000x50,000 to 390x390x128, which fits into GPU memory.

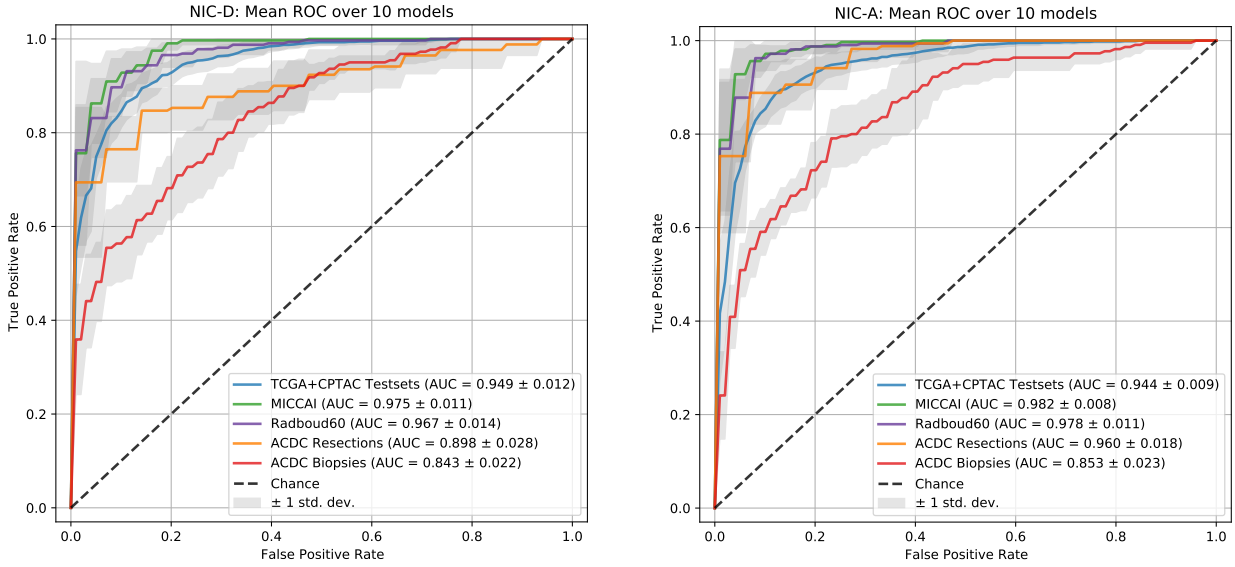
While the initial version of NIC[8] used encoders trained in an unsupervised way, it was recently shown[13] that an encoder trained to classify patches for multiple tasks (not related to the slide-classification task) resulted in learning a better representation of the histopathology image data. In this paper, we use this multi-task encoder for compressing the lung slides. The encoder was trained on overall 800,000 patches of size 128x128 for four tasks: axillary lymph node tumor metastasis detection, mitosis detection in breast, prostate epithelium detection, and colorectal cancer tissue type classification. Note that no lung tissue was used to train the encoder. Details of the encoder and the used datasets can be found in[13] and.[14] The encoder and code examples for its usage are publicly available.[15]

For the classifier CNN we evaluated two architectures: A small DenseNet[16]-like model (NIC-D) and the slightly adapted attention-architecture of CLAM (NIC-A). NIC-D (see Fig. 3a) consists of a single DenseBlock with three layers having growth rate 32 and dropout. Before the final sigmoid layer, global max pooling is applied. The first layer in the network was set to a 3x3 convolution with ReLU activation. The classifier has overall seven convolutional layers and 145 thousand trainable parameters, half of them in the first layer. NIC-A (see Fig. 3b) consists of a convolutional layer, followed by attention-pooling (attention-weighted sum of all encoded patches) and a classification layer. The difference to the original CLAM-classifier architecture is the addition of the first layer and the reduction of the number of features. The classifier has overall 181 thousand trainable parameters, most of them (147 thousand) in the first layer.

We trained the classifiers on the encoded slides from CPTAC and TCGA. The slides were encoded with flip- and rotation-augmentations resulting in 8 encodings per slide. While an encoded slide fits into memory, fitting many slides at once as part of a training mini-batch is still not possible. To increase the batch size, during training, crops of size 200x200x128 were taken, which approximately corresponds to the average compressed

Model	TCGA+CPTAC	MICCAI	Radboud60	ACDC resections	ACDC biopsies
NIC-D	**0.949 ± 0.012**	0.975 ± 0.011	0.967 ± 0.014	0.898 ± 0.028	0.843 ± 0.022
NIC-A	0.944 ± 0.009	**0.982 ± 0.008**	**0.978 ± 0.011**	**0.960 ± 0.0218**	**0.853 ± 0.023**

Table 2: AUC values assessed on the test datasets.



(a) NIC-D **(b)** NIC-A

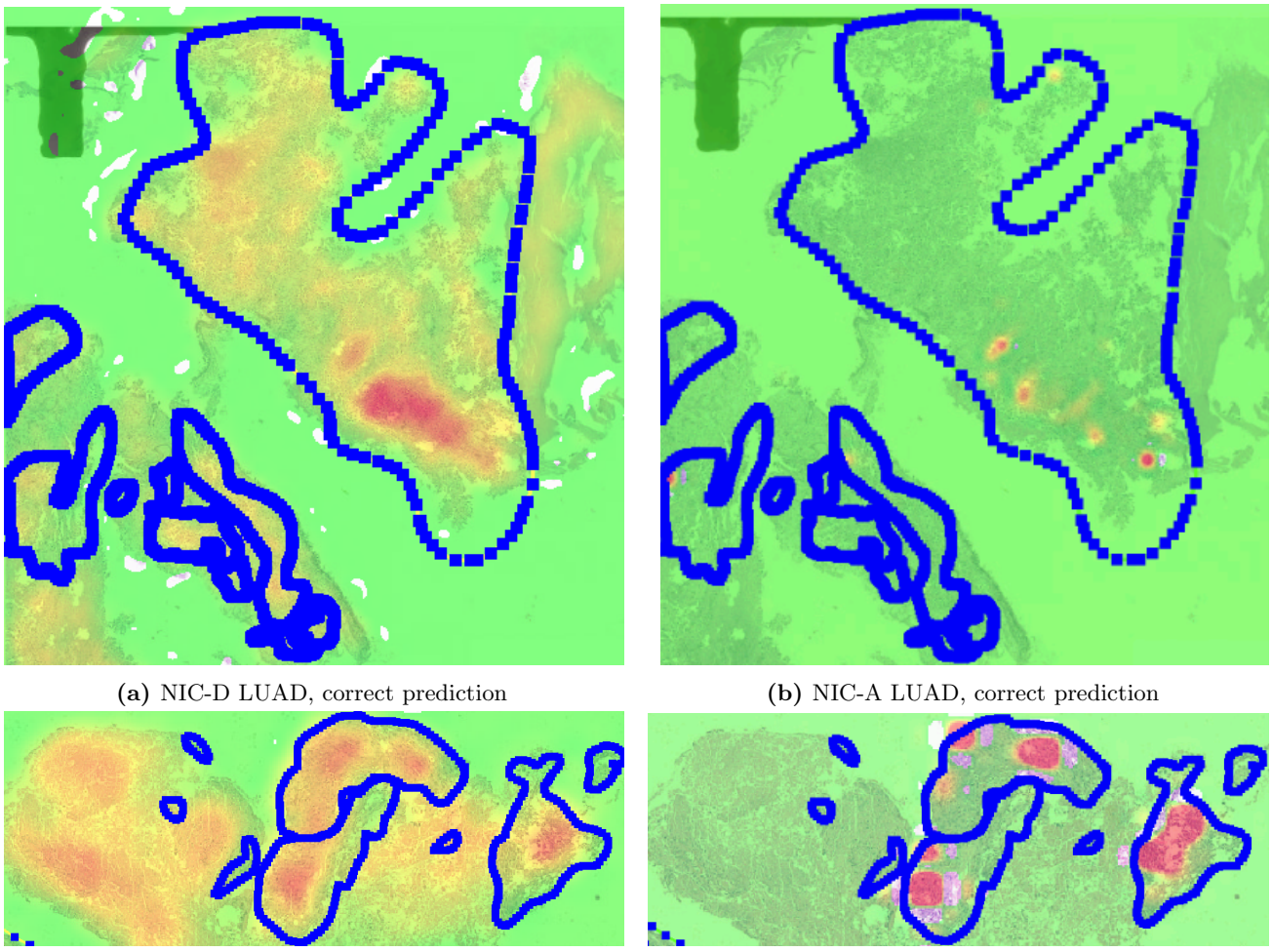
Figure 3: Mean ROC over 10 models trained each for CPTAC+TCGA split with NIC-D (left) and NIC-A (right). 'Positive' refers to the LUAD class.

slide size in the TCGA+CPTAC dataset. This can be also seen as a form of crop augmentation. Since lung cancer often covers large slide areas, most crops contain cancerous tissue. During testing, the complete slides were processed (made possible by fully convolutional layers and global pooling). For training we used the cross-entropy loss with the Adam optimizer. The ROC-AUC score on the validation set was used to reduce the learning rate on plateaus after 10 epochs and to stop early when there was no more improvement after 30 epochs. Finally, the network with the highest validation AUC was selected.

4. RESULTS

During training the models usually converge to 0.99 training AUC after about 60 epochs. The resulting average receiver operating characteristic (ROC) curves and corresponding area under the curve (AUC) values for the ten models on the TCGA+CPTAC testsets and the other datasets are depicted in Figure 3 and Table 2. Both classifier architectures achieve AUC values >0.94 on most datasets. NIC-A outperforms NIC-D on all external datasets and is only slightly worse on the TCGA+CPTAC testset.

To better understand the inner workings of the networks we visualized what the networks 'look' at. For NIC-A this is the attention-heatmap, for NIC-D we use gradcam,[17] a technique to highlight parts of the input responsible for the networks decision. Fig. 4 shows the NIC-D gradcam and the NIC-A attention heatmaps for a LUAD and a LUSC case from ACDC. In the LUAD case the prediction of both models was correct. In the LUSC case, NIC-D wrongly predicted the case to be LUAD while the prediction of NIC-A was correct. The comparison of the heatmaps shows that the attention of NIC-A is much more sparse than the gradcam of NIC-D. NIC-A selects only a few patches and ignores the other. NIC-D focuses on more patches, but not all tumor-patches (inside the blue polygon-annotations). It also focuses on non-tumor tissue, in other cases even more then on tumor-tissue. This might be responsible for the wrong prediction for the shown LUSC slide, where beside activations in the tumor-regions also activations in wide areas with i.e. necrosis occurred.



(a) NIC-D LUAD, correct prediction

(b) NIC-A LUAD, correct prediction

(c) NIC-D LUSC, wrong prediction

(d) NIC-A LUSC, correct prediction

Figure 4: Two cases from ACDC overlayed with the gradcam and attention heatmaps of the NIC-D (left) and NIC-A models (right). The blue contours are the pathologist' tumor annotations.

5. DISCUSSION

Our models achieve AUCs >0.94 on the TCGA+CPTAC testsets, the MICCAI and the Radboud60 cohorts. The performance on the ACDC resections is slightly lower. This might indicate that this cohort is more different from the TCGA+CPTAC data than the other datasets. Despite the lower AUC value of 0.84-0.85 on the ACDC biopsies, this shows the applicability of the models solely trained on resections to biopsies, albeit with a performance drop of 5-10%. This can be improved in the future by adding cases of biopsies to the training set. Except for ACDC, there is no drop in performance when applying the model to the independent, external cohorts, suggesting that the model is mostly center-independent. The reason for such a robustness might be attributed to the encoder, which was trained on histo-pathological multi-centric data, implicitly providing a certain degree of invariability regarding color staining variations. It is hereby of note that the encoder was not trained on lung data and also not to differentiate between different cancer types. Nevertheless, the encoding experimentally showed to contain enough information to discriminate between LUAD and LUSC.

The attention-based architecture of NIC-A performs better on the evaluation datasets then the feature-oriented architecture of NIC-D. This indicates that learning simple features from a few attended patches generalizes in this case better then trying to learn more complex features. If the encoder provides a good enough representation, more complex features might not be necessary and lead to a brittle classifier. Since for training only two thousand labels are used, the biggest danger is overfitting. The main measures against it in our approach are the encoding-augmentations and low classifier complexity with less then 200,000 parameters. A further reduction of parameters led to a decreased performance in preliminary experiments.

The gradcam and attention visualizations revealed very different attention patterns in the architectures despite relative similar performance results. This difference is probably due to the global max-pooling in NIC-D and attention pooling in NIC-A. Both pooling operations endorse attention sparseness, since the network can concentrate on a few features and patches instead of having to include all patches into the decision as in the case of average pooling. In the optimal case we would expect the models to focus on the cancer areas and ignore everything else. This did not happen here and might require additional model adaptations and constraints if set as a goal. An interesting scenario would be to try to combine a patch-level classifier when some annotations are available with a slide-level classifier. In NIC, this is currently only possible indirectly by using the patch-level classifier as encoder (without the final classification layers).

NIC can be seen not only as a concrete architecture, but also as a framework encompassing all methods working on compressed slides. From that perspective CLAM could be also considered as an instance of NIC with additional assumptions and constraints on patch-level learning. Both methods require all patches of a slide to be loaded into memory and could not work on uncompressed patches (without severe technical adaptations or very large memory).

NIC approaches the state-of-the-art performance of the recently presented CLAM method (reported AUC of 0.956 on their TCGA+CPTAC test-splits) without imposing as many assumptions on the data. The NIC-D model uses a tiny DenseNet classifier with good results. They are improved upon with the NIC-A model, which incorporates the attention-weighted pooling of CLAM, but without using attention-based clustering. The NIC-framework can be adapted to the task at hand with, i.e. attention, while keeping its flexibility for larger spatial pattern recognition. Additional assumptions on the data, such as the number of patches important for classification are not imposed. A direct methodical comparison to the full CLAM method is made difficult by different preprocessing steps and different encoders. The encoder used in this work was trained on histopathological images with a relative high compression factor of 384x (from 128x128x3 patches to a 128-length vector) while CLAM uses an ImageNet-pretrained encoder with a compression factor of 192x (from 256x256x3 patches to a 1024-length vector). In future work we want to analyse the individual contributions of the encoder and the classifier to the performance, carry out a thorough comparison between NIC and CLAM and extend the training set and the evaluation to a larger number of lung biopsies.

6. ACKNOWLEDGMENTS

Part of the data used in this publication was generated by the National Cancer Institute Clinical Proteomic Tumor Analysis Consortium (CPTAC). The project was partly funded by the Dutch Cancer Society (KWF) within the PROACTING project No 11917 and European Union's Horizon 2020 research and innovation programme

under grant agreement No 825292 (ExaMode, http://www.examode.eu/). This work has not been submitted for publication or presentation elsewhere.

REFERENCES

1. A. C. Society, "Lung cancer statistics." `https://www.cancer.org/cancer/lung-cancer/about/key-statistics.html`, Accessed in August 2020.
2. A. C. Society, "What is lung cancer?." `https://www.cancer.org/cancer/lung-cancer/about/what-is.html`, Accessed in August 2020.
3. G. Litjens, T. Kooi, B. Ehteshami Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," vol. 42, pp. 60–88, 12 2017.
4. L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, and J. H. Saltz, "Patch-based convolutional neural network for whole slide tissue image classification," in *Proceedings of the ieee conference on computer vision and pattern recognition*, pp. 2424–2433, 2016.
5. N. Coudray, P. S. Ocampo, T. Sakellaropoulos, N. Narula, M. Snuderl, D. Fenyö, A. L. Moreira, N. Razavian, and A. Tsirigos, "Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning," *Nature medicine*, vol. 24, no. 10, pp. 1559–1567, 2018.
6. G. Campanella, M. G. Hanna, L. Geneslaw, A. Miraflor, V. W. K. Silva, K. J. Busam, E. Brogi, V. E. Reuter, D. S. Klimstra, and T. J. Fuchs, "Clinical-grade computational pathology using weakly supervised deep learning on whole slide images," *Nature medicine*, vol. 25, no. 8, pp. 1301–1309, 2019.
7. M. Y. Lu, D. F. Williamson, T. Y. Chen, R. J. Chen, M. Barbieri, and F. Mahmood, "Data efficient and weakly supervised computational pathology on whole slide images," *arXiv preprint arXiv:2004.09666*, 2020.
8. D. Tellez, G. Litjens, J. van der Laak, and F. Ciompi, "Neural image compression for gigapixel histopathology image analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
9. K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, *et al.*, "The cancer imaging archive (tcia): maintaining and operating a public information repository," *Journal of digital imaging*, vol. 26, no. 6, pp. 1045–1057, 2013.
10. J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. M. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, J. M. Stuart, C. G. A. R. Network, *et al.*, "The cancer genome atlas pan-cancer analysis project," *Nature genetics*, vol. 45, no. 10, p. 1113, 2013.
11. 20th International Conference on Medical Image Computing and C. A. Intervention, "Computational precision medicine challenge," 2017.
12. "Automatic cancer detection and classification in whole-slide lung histopathology (acdc@lunghp)." `https://acdc-lunghp.grand-challenge.org/Challenge`. Accessed in August 2020.
13. D. Tellez, D. Hoppener, C. Verhoef, D. Grunhagen, P. Nierop, M. Drozdzal, J. van der Laak, and F. Ciompi, "Extending unsupervised neural image compression with supervised multitask learning," in *Medical Imaging with Deep Learning*, 2020.
14. D. Tellez, G. Litjens, P. Bándi, W. Bulten, J.-M. Bokhorst, F. Ciompi, and J. van der Laak, "Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology," *Medical image analysis*, vol. 58, p. 101544, 2019.
15. D. Tellez and W. Aswolinskiy, "Neural imgae compression multi-task encoder github repository." `https://github.com/davidtellez/neural-image-compression`, 2021.
16. G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.
17. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.