PHILIPS

Computational pathology

Technology backgrounder

Seamless Computational Pathology

Dr. Bas Hulsken, CTO Philips Digital Pathology Solutions

Around the world, digital pathology is unchaining pathologists from their (physical) labs, allowing them now to practice anatomic pathology anywhere, independent of the physical location of the biopsy, the glass slide and the patient. The tremendous acceleration of digital pathology coincides with a rapid increase in the capabilities as well as the applications of artificial intelligence.

Analogous to professions that have previously made the move to digital, these two technology revolutions have merged to create a new field: computational pathology. This creates new opportunities for assisting pathologists and pathology labs in improving diagnostic efficiency and quality. We are at the brink of converting local pathology labs into world-encompassing virtual pathology networks in which computer intelligence will seamlessly team up with pathologists. This will help meet challenges presented by aging populations, increasing targeted therapies and rising shortage of pathologists.

In this white paper, I will explain how Philips intends to deliver computational pathology to the clinical pathology lab through streaming, deep learning-based image analytics, that is seamlessly embedded in the digital pathology workflow, with the aim to enhance pathologists' reading and reporting with rich, relevant data that are immediately available, without delay, when a case is first opened.

1. Computational Pathology

It is hard to overstate the efficiency and accuracy a pathologist requires to read a case and arrive at an accurate diagnosis. Perhaps the most important thing to get right with computational pathology is to not get in the way of the pathologists, who possess all the

knowledge and skills required to read a case. When and where computer intelligence can add value, it is essential that it supports pathologists with the smallest possible delay and interaction requirement. The information provided should be easy to accept, reject or amend by the pathologist. This is all geared towards assisting the pathologist in increasing diagnostic efficiency and quality, while enhancing and enriching the pathology report with relevant, quantitative data. Areas in which computational pathology can help the pathologist in a meaningful way are:

1. **Quantification** of cells, structures or stain intensities, e.g. mitotic count, IHC stain quantification, tumor area measurement, or tumor cell percentage determination.
2. **Localization** of relevant morphologies, such as the localization of (micro-) metastases, tumor, necrotic tissue, lymphocytes, or the segmentation of epithelial versus stroma.
3. **Classification** of pathologies, for instance tumor grading with the Nottingham or Gleason score, or determination of invasive versus in-situ.

For each of the three categories mentioned above, impressive examples of computational pathology algorithms can be found in literature. Examples include the detection of (micro) metastases in lymph nodes,[1] the detection of prostate tumor,[2] the quantification of breast panel[3] and the automated annotation of tumor for molecular pathology and genomic profiling.[4]

And yet, computational pathology plays no role of significance in the practice of clinical pathology today. Why is this the case? The answer is implied in the opening sentences of this paragraph: today's computational pathology applications get in the way of the

pathologist. The few computational pathology applications that have made it beyond the research stage require significant interaction from the pathologist. They might require tens of seconds or even minutes to compute a result, and are not seamlessly embedded in the pathology workflow. This confines today's computational pathology applications to areas in which the increase in diagnostic quality is large enough to compensate for the enormous decrease in efficiency. Unless we can find a way to change this, the full potential of computational pathology will not be realized.
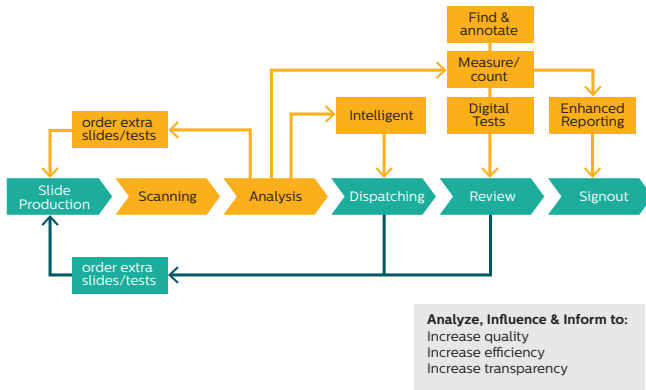

Figure 1: Schematic overview of a typical workflow in a pathology lab enhanced with seamlessly integrated computational pathology 'apps' (orange). The workflow is organized around the pathologist performing case reviews. Seamless integration of computational pathology is realized by executing the relevant 'apps' on suitable whole slide images directly after scanning, and without any user interaction, on the streaming Analytics Engine (orange). The (combined) results of the relevant computational pathology apps are instantly available upon opening the case for review.

2. Computational Pathology Workflow

To remedy the efficiency problem inherent in today's computational pathology applications, we may take inspiration from the way digital pathology is currently transforming the pathology lab. In the past, digital pathology solutions required significant operator interaction to create a digital image of a slide. Selection of the region of interest, magnification and focus points were followed by a manual quality check of the scanned image and, if necessary, a rescan. Today, fully automated, interaction-free scanners are enabling the successful digital transformation of pathology labs. The secret to full automation is 'Do More!' – the entire tissue slide is digitized at the highest possible quality and resolution, fully automatically, irrespective of the region and image magnification the pathologist might require to review the slide. After the relatively slow, but fully automated, process of slide scanning is complete, the pathologist decides, in the fast interactive image viewer, which parts of the image to view. In practice, this amounts to less than 1% of the image data on average.

Fig. 1 shows how an equally efficient computational pathology workflow can be achieved by applying the same principle. The

relatively slow process of running computational pathology algorithms on whole slide images is fully automated, and the pathologist can decide which pre-calculated results from the computational pathology apps to use (and which to ignore) when first reviewing the case. This seamless workflow is most efficiently implemented by running all relevant computational pathology apps during or immediately after scanning the slide. This approach seamlessly joins the two most time-consuming procedures in the computationally enabled digital pathology lab into a single fully automated 'scan and analyze' process. The result: immediate availability of maximum quality whole slide images enhanced with full data analysis results from all relevant computational pathology apps when the pathologists first open a case for review.

3. Computational Pathology Apps

The seamless computational pathology workflow proposed in the previous paragraph relies on the ability to easily plug in multiple diverse and powerful computational pathology apps. The Philips IntelliSite Pathology Solution[5] is built on a scalable, secure, web oriented platform. Its full functionality is enabled through rich internet applications that run directly in the browser. This mandates that the computational pathology apps themselves do not do any of the computational work and cannot be relied upon to drive the workflow. Instead, they run in the browser, on a low-powered workstation or even a mobile device that may lose its connection to the platform at any moment. At the same time, an effective computational pathology platform should allow for easy deployment of diverse, powerful and self-contained computational pathology apps. To deliver both aforementioned items,
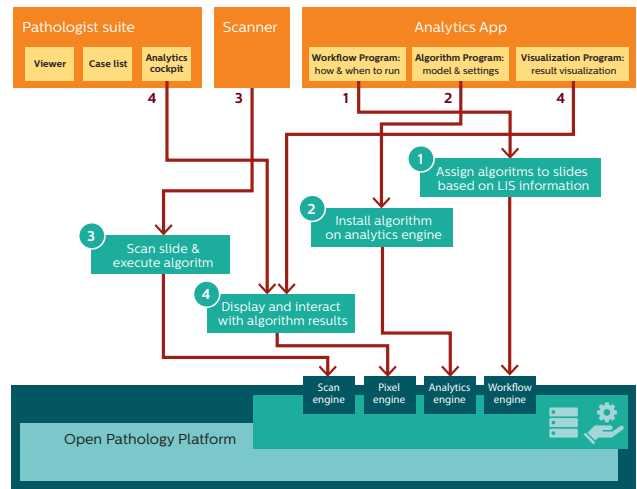

Figure 2: Architecture of a computational pathology app and platform. All work is done by the platform, but the app contains all the 'intelligence'; it defines the algorithm, on which slides to run it, and how to present the results to the pathologist. This upfront programming consists of two steps: 1) select applicable slide types (stain, organ, etc.), and 2) program the app's tasks on the analytics engine. These tasks are performed automatically in step 3) when a relevant slide is scanned. Finally when a pathologist reviews a relevant case, step 4) displays and allows interaction with the pre-calculated results.

computational pathology apps need to be designed as shown in fig. 2: smart apps that contain all the 'intelligence' but rely on the underlying platform for the required computing power and reliable workflow execution. The computational pathology app needs to be designed as an easily deployable package containing all of its specific algorithms, workflow rules and visualization methods, running on the platform's programmable analytics and workflow engines.

4. Automated Complex and Simple Interactive Tasks

An essential requirement for any computational pathology app is offering pathologists complete freedom to easily accept or reject the results of the app, and also be able to adjust and combine them. For example, an app that combines tumor segmentation with a mitotic count must allow the pathologist to adjust the detected perimeter of the tumor and be immediately informed of the tumor's updated mitotic count. Another example: adjusting the sensitivity and specificity thresholds for a metastasis detection app should instantaneously update the visualization and count of all metastases found in the image. This interactivity can be effectively achieved through the previously introduced 'Do More!' concept: finding the mitotic figures with, for instance, a neural network in a whole slide image is a complex and computationally intensive task, as is finding all tumor areas. However, these tasks, when applied indiscriminately to the whole slide image, can be performed upfront in a fully automated manner. As a result, the pathologist won't need to wait for their completion. The task of counting the already detected mitotic figures in a given tumor region, on the other hand, needs to be interactive: it should respond to the pathologist revising the tumor perimeter. These 'simpler' tasks can, however, be performed at adequate speed, once again so the pathologist does not need to wait. Thus, to create powerful and responsive computational pathology apps, we need to break them down in automated complex tasks and simple interactive tasks, as shown in fig.3.

5. Scalable Streaming Analytics

With the computational pathology workflow and apps described in the previous paragraphs, we have all the ingredients for a functional and seamless deployment of computational pathology. However, when it comes to successful adoption, cost and performance are also to be considered. Computational pathology aims to bring considerable added value to the pathologist. However, as a consequence of that success, many computational pathology apps will inevitably be applied to a significant percentage of generated whole slide images. To arrive at an optimal cost/performance ratio, Philips IntelliSite Pathology Solution will be designed as 'scale-out' architecture (fig. 4). This architecture does not rely on the increasingly difficult challenge of making a single unit perform faster and faster, but instead relies on distributing the work over multiple identical units working in parallel. The absence of bottlenecks – such as situations in which a single unit has to do all the work – is essential to this type of scale-out architecture. In computational pathology, image storage may easily become a bottleneck. We would like a single image management solution to contain all cases, however, if all scanners were to transmit their data to a single storage system, and all algorithms would read from and write to that same storage system, the load of that system would be effectively doubled or even tripled, if the algorithms generate significant amounts of data themselves.

To arrive at a truly scalable computational pathology workflow in the clinical pathology lab, the complex automated tasks required by computational pathology apps need to be performed before the whole slide images are send to the storage system. The Philips IntelliSite Pathology Solution aims to achieve this by leveraging Philips' iSyntax image streaming technology[6] to route and branch image streams from one or more scanner(s) as multiple parallel streams towards the computers running the algorithms and to the image management system. The simple interactive tasks can subsequently be performed when triggered by the pathologist in the pathologist suite. This approach requires a programmable streaming analytics engine.
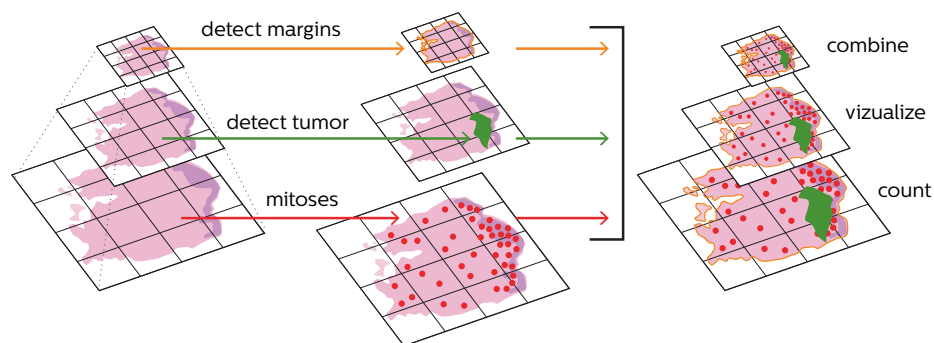

Figure 3: Method for breaking down computational pathology apps into computationally intensive tasks, such as finding mitoses with deep learning, and simpler interactive tasks, such as counting found mitoses in a detected tumor area. The complex tasks run indiscriminately on the whole slide, while the pathologist triggers the interactive tasks. The interactive tasks could encompass real-time adjustable advanced visualization tools to augment the tissue image with image analytics results to enhance the pathologists' image interpretation and to drive adoption into routine clinical practice.
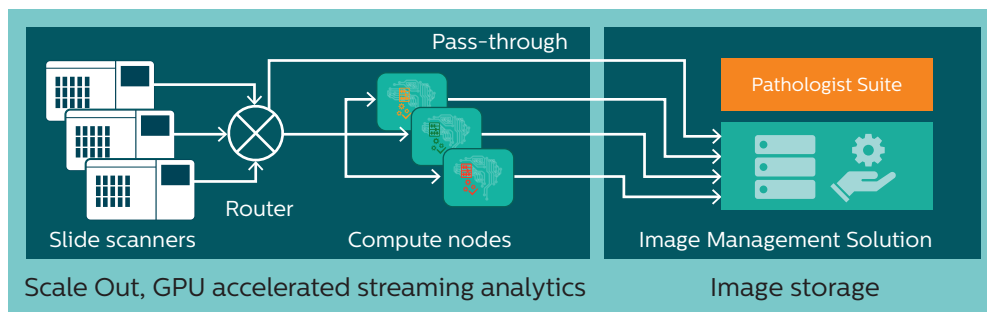
Figure 4: Scalable streaming analytics by routing multiple parallel image streams from the scanner(s) to multiple computers running image analytics tasks in parallel and to the image management system.

6. Open Analytics

By exploiting iSyntax' wavelet based image streams, which can be accessed efficiently at any magnification level, the future design of the Philips IntelliSite Open Pathology Platform's streaming analytics engine will efficiently run each image analytics task at its optimal magnification. It will be possible to build apps for the platform in an open, well-documented manner. For example with neural networks trained on the various (open) deep learning frameworks. The easy addition of third party apps to the pathology workflow is essential for computational pathology. This will only flourish as an ecosystem grows around it. Knowledge, the key ingredient in any computational pathology app, is found in pathologists all over the world. That knowledge can only be harvested through collaboration.

Computational pathology apps can support the pathologist by improving diagnostic accuracy through, for example, quantification of IHC stains, or detection of metastases, but also by increasing efficiency through streamlined workflows, for example, case prioritization and automated dispatching, automated ordering of tests, and more. Myriads of similar computational apps can be imagined and indeed are necessary for creating a computational pathology-enabled lab. We need to invest in building the ecosystem, since no single party could provide all of these apps, or even enough of them to provide computational pathology with the critical mass it needs to deliver its full promise.

References

[1] D. Wang, A. Khosla, R. Gargeya, H. Irshad, A.H. Beck: *Deep Learning for Identifying Metastatic Breast Cancer*, arXiv:1606.05718 [q-bio.QM] (2016)

[2] G. Litjens, B. Ehteshami Bejnordi, N. Timofeeva, G. Swadi, I. Kovacs, C.A. Hulsbergen-van de Kaa and J.A.W.M. van der Laak: *Automated detection of prostate cancer in digitized whole-slide images of H and E-stained biopsy specimens*, Proc. SPIE 9420, Medical Imaging 2015: Digital Pathology, 94200B (2015)

[3] G. Stalhammar, N. Fuentes Martinez, M. Lippert, N.P. Tobin, I. Mølholm, L. Kis, G. Rosin, M. Rantalainen, L. Pedersen, J. Bergh, M. Grunkin and J. Hartman: *Digital image analysis outperforms manual biomarker assessment in breast cancer Modern Pathology* 29, 318- 329 (2016)

[4] P. W. Hamilton, Y. Wang, C. Boyd, J. A. James, M. B. Loughrey, J. P. Hougton, D. P. Boyle, P. Kelly, P. Maxwell, D. McCleary, J. Diamond, D. G. McArt, J. Tunstall, P. Bankhead, M. Salto-Tellez: *Automated tumor analysis for molecular profiling in lung cancer Oncotarget* 6 27938- 27952 (2015)

[5] B. Hulsken: *Philips IntelliSite Open Pathology Platform, white paper* (2016).

[6] B. Hulsken: *Philips' iSyntax for Digital Pathology, white paper* (2016).