Computer Methods and Programs in Biomedicine 179 (2019) 104983

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Optimum web viewer application for DICOM whole slide image visualization in anatomical pathology

Nieves Lajara, Jose Luis Espinosa-Aranda, Oscar Deniz, Gloria Bueno*

VISILAB, University of Castilla-La Mancha, Ciudad Real, Spain

ARTICLE INFO

Article history:
Received 15 February 2019
Revised 20 June 2019
Accepted 10 July 2019

Keywords:
DICOM Web viewer
Digital pathology
Whole-Slide imaging
Web viewer
Microscopic image visualization
Anatomical pathology DICOM connectathon

ABSTRACT

Background and Objective: Digital scanners are being increasingly adopt-ed in anatomical pathology, but there is still a lack of a standardized whole slide image (WSI) format. This translates into the need for interoperability and knowledge representation for shareable and computable clinical information. This work describes a robust solution, called *Visilab Viewer*, able to interact and work with any WSI based on the DICOM standard.

Methods: Visilab Viewer is a web platform developed and integrated alongside a proposed web architecture following the DICOM definition. To prepare the information of the pyramid structure proposed in DICOM, a specific module was defined. The same structure is used by a second module that aggregates on the cache browser the adjacent tiles or frames of the current user's viewport with the aim of achieving fast and fluid navigation over the tissue slide. This solution was tested and compared with three different web viewers, publicly available, with 10 WSIs.

Results: A quantitative assessment was performed based on the average load time per frame together with the number of fully loaded frames. Kruskal–Wallis and Dunn tests were used to compare each web viewer latency results and finally to rank them. Additionally, a qualitative evaluation was done by 6 pathologists based on speed and quality for zooming, panning and usability. The proposed viewer obtained the best performance in both assessments. The entire architecture proposed was tested in the 2nd worldwide DICOM Connectathon, obtaining successful results with all participant scanner vendors.

Conclusions: The online tool allows users to navigate and obtain a correct visualization of the samples avoiding any restriction of format and localization. The two strategical modules allow to reduce time in displaying the slide and therefore, offer high fluidity and usability. The web platform manages not only the visualization with the developed web viewer but also includes the insertion, manipulation and generation of new DICOM elements. *Visilab Viewer* can successfully exchange DICOM data. Connectathons are the ultimate interoperability tests and are therefore required to guarantee that solutions as *Visilab Viewer* and its architecture can successfully exchange data following the DICOM standard. Accompanying demo video. (Link to Youtube video.)

1. Introduction

Telepathology facilitates remote access to specialist expertise, which in turn results in better and often more efficient patient care [1]. This practice of pathology at a distance is directly related to advances in computing technology and the Internet. However, the use of telepathology for clinical patient care has been restricted mostly to large academic organizations. Obstacles that have limited its widespread use include prohibitive costs, legal and regulatory issues, technologic drawbacks, obstinacy from pathologists, and above all a lack of universal standards [2].

Technological improvements allow manufacturers to build enhanced microscopy scanners to obtain better whole slide imaging (WSI) quality. Robust workflows ensure reliable and accurate acquisitions that cover steps such as sectioning, staining, scanning and storage of tissue samples [3].

Methodologies and concepts from the radiology domain are adopted to assemble collaborative Digital Anatomic Pathology environments. Once scanned and acquired, the WSI slides are stored and managed by a picture archiving and communication system (PACS) normally installed inside the specific clinical pathology department in the health-care facility. In order to achieve an inter-

* Corresponding author.
E-mail address: Gloria.Bueno@uclm.es (G. Bueno).

https://doi.org/10.1016/j.cmpb.2019.104983
0169-2607/

operable clinical information system, some international standards of medical informatics are recommended, such as DICOM (Digital Imaging and Communications in Medicine).

The DICOM pathology working group 26 (WG26) created in 2005, formalizes DICOM information object definition applicable to WSI slides. Two new DICOM supplements were defined by the WG26: a) supplement 122 to define the use of specimen identification attributes to support the imaging workflow in pathology departments [4]; b) supplement 145 to describe WSI characteristics to be stored into PACSs using DICOM-standard messaging [5]. The storage provisions made in DICOM supplement 145 is quite significant since it conditions the design of the software dedicated to visualizing the WSI slides.

Despite the formal definition of an international standard such as DICOM, the use of proprietary software and devices with closed formats prevails. Even remote work depends on using remote desktop applications over VPN as exposed in [3]. Once the particular manufacturer scanner digitizes glass slides the data normally must be accessed and visualized with the same manufacturer-specific software to interpret the vendor format. These factors represent the main limitations of telepathology and collaborative work. Furthermore, not all scanners have software that easily integrates with laboratory information systems (LIS) [6].

The DICOM standard is not widely addressed yet. Pathologists' demand addresses how global standardization is adopted among manufacturers and scientists. Furthermore, each specialization laboratory has its own technique for staining, sectioning, imaging, storing and even scoring the slides [7]. Commercial solutions usually create their own proprietary methods. Combining technology, equipment and software with other vendors is not feasible and physicians are not able to collaborate during diagnosis. Only by introducing web platforms based on universal standards pathologists' reluctance will decrease and their requirements for standards will increase [8]. The promotion of DICOM will make scanner vendors focus on interoperability. This would be most desirable for telepathology networks involving WSI platforms from multiple vendors [9].

The shipping of slides presents an obvious risk of losing data and privacy [10]. Consultation and knowledge sharing need for a suitable framework. Previous studies already proposed a standalone web solution to visualize the WSI inside a proper DICOM architecture and environment [11]. Nevertheless, very strong limitations and drawbacks cannot be avoided due to the closed test environment, which is tied to a particular scanner and PACS server. There is a formal requirement to precisely test any DICOM slide from any manufacturer to prove interoperability according to the standard.

In this paper, an efficient and scalable solution that provides compatibility and interoperability based on DICOM is proposed. This solution was tested in the two first DICOM Connectathons ever held for anatomical pathology, promoted and organized by the WG26, obtaining excellent results [12]. A Connectathon is an event that is centered on an open consensus built Interoperability (Connection) definition. The purpose of a Connectathon is both to prove that the specification is complete as well as to prove that implementations written to conform to that consensus can 'connect'.

In healthcare, the Integrating the Healthcare Enterprise (IHE) promotes the progress of Connectathons and the DICOM WG26 promotes these events for digital pathology.

The entire work presented here consists of the integration of various modules including a PACS, a DICOM pyramid web viewer named *Visilab Viewer* and secondary server interfaces that optimize data management. Quantitative and qualitative experiments were carried out to evaluate visualization performance.

2. Material and methods

The first milestone covered the development of the web viewer in parallel with the middleware web service layers. Performance was tested in a real industrial context in the first WSI DICOM Connectathon. Later, a PACS storage manager was integrated to fulfill the continuous WSI data provider module based on DICOM. In a second Connectathon, the solution was provided as the main DICOM repository where several scanner vendors uploaded their acquisitions in real time.

The data obtained during the Connectathons was used to estimate, quantitatively and qualitatively the performance of the implemented web viewer. Six independent pathologists participated in a set of double-blind experiments that consisted of examining 10 WSIs from different tissue samples.

2.1. Materials

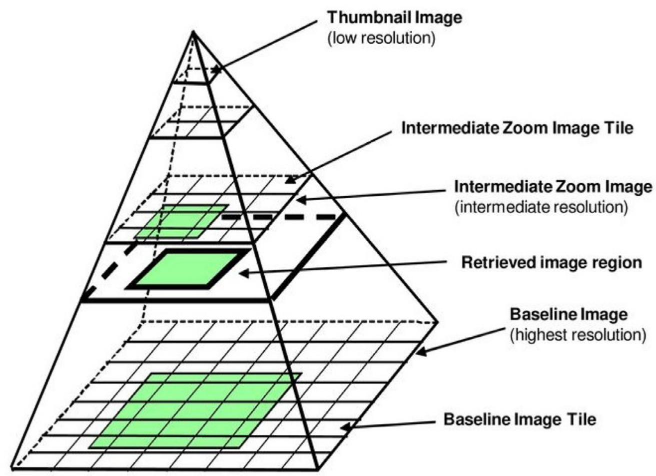
Ten WSIs were obtained from the digitalization of fine needle aspiration biopsies and cytologies obtained from different tissues and with different scanners. As a pyramid addresses zooming, the usual organization of a WSI may be thought of as a pyramid. Image data may be stored in a tiled pattern to achieve rapid panning. Thus, this pyramidal structure facilitates fast retrieval of regions of interest from the full WSI, as shown in Fig. 1a.

The key mechanism for storing DICOM WSI is to store the individual tiles of a single WSI pyramid level as individual frames in a DICOM multi-frame image object as shown in Fig. 1b [5]. A tile may be fetched with a unique reference number indicating its distinctive level, *x* and *y* coordinates inside the pyramid.

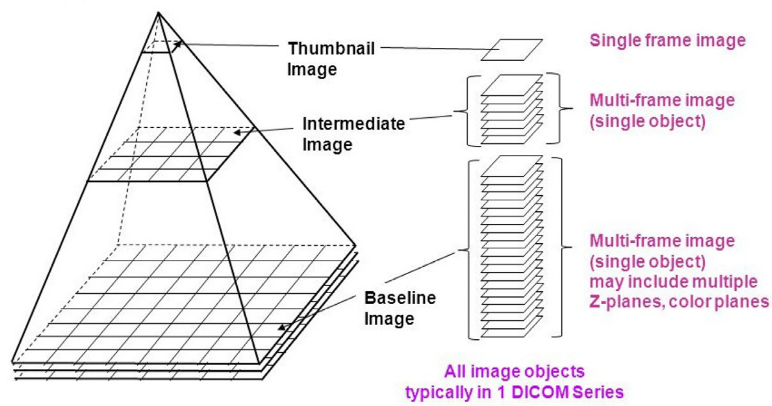
Fetching, rendering and final visualization may involve extreme processing for the web viewers. Therefore the slides selected in the experiments are meant to test different issues, for example, missed levels in the DICOM pyramid, a large number of frames, or huge dimensions of the slide. The behavior of the viewer to these issues will also determine the user experience, and allows for a comparative study based on the loading latency of pyramid frames. Table 1 shows the properties of the slides used in the experiments.

Table 1
4 DICOM Whole Slide Images.

Slide	Type	Manufacturer	#Frames	Frame Size (pixels)	WSI Size
A	Cervical Citology	Ventana	4096	961 × 1119	3.82GB
B	Subcutaneous Skin	Hamamatsu	48	1024 × 1024	34.25MB
C	Mammary Gland	Hamamatsu	48	1024 × 1024	42.61MB
D	Bone & Soft Element	Hamamatsu	3672	1024 × 1024	2.31GB
E	Skin	Aperio	4096	479 × 468	502.94MB
F	Fungi	Ventana	120	1024 × 1024	190.18MB
G	Colon Polyps	Hamamatsu	224	1024 × 1024	104.75MB
H	Placenta	Hamamatsu	2112	1024 × 1024	2.17GB
I	Liver	Aperio	4096	2740 × 2022	631.37MB
J	Papillary Bladder	Aperio	16,384	934 × 620	7.82GB



(a) Whole slide image as a "Pyramid" of image data.



(b) Correspondence of an image pyramid to DICOM images and series.

Fig. 1. Description of WSI organization and structure proposed by DICOM WG26 [5].

2.2. PACS DICOM Architecture

The design and integration of the architecture components are depicted in Fig. 2. From the point of view of the final user, it is only necessary to fetch the web application and start to query data consisting of the slides in DICOM format. Queries are not directly sent to the PACS over the DICOM messaging protocol, but instead, they initially use an intermediate web server interface between the PACS and *Visilab Viewer*. The purpose of this strategy is to provide scalability in terms of connectivity to other architectures or PACS systems.

The proposed DICOM viewer contains a panel configuration whereby users can set the PACS system to use. This panel includes options to define the endpoints of the DICOM query calls defined by the standard. Users may extract data from any PACS that is compliant with the DICOM protocol.

Client viewer queries will travel across two layers before reaching the PACS system. The first layer manages the main services between *Visilab Viewer* and the remaining system where the raw data is stored. If using the proposed PACS system architecture, a second layer is provided with its own API Management. This layer contains the functionality to convert proprietary format slides into DICOM objects. New slides can be uploaded by the user from the web viewer. The results of their conversions into DICOM format are also stored in the PACS at the back-end layer.

Each frame number inside an instance refers to a particular tile inside the pyramidal structure. This information is not stored outside the PACS and therefore it is not directly available. The web client previously needs to understand the internal organization of the slide within a frameset structure.

The architecture design enables participants in the Connectathons to interact directly with the last layer where the PACS resides, skipping the intermediate layers. Their associated scanners communicate, store onto and fetch data from the PACS provided in the solution suggested. Most of the vendor participants are widely regarded as world-leaders in the field (they are shown in Fig. 4b). VISILAB represents the system architecture capable of managing the storage of DICOM slides from different scanners in real time and also provides the *Visilab Viewer* client to inspect and visualize the raw data acquired.

2.3. Visilab viewer

The proposed visualization method follows the pyramidal structure defined in DICOM supplement 145 [5]. The rendering of each subregion is presented in a similar way as widely-used browser-based map viewers. Currently the supported formats are:.tif,.svs,.ndpi,.mrxs,.jp2 that may be converted to pyramidal.tif or.dcm. The development of *Visilab Viewer* is based on the OpenSeadragon source map browser [13] and evolved in customization that includes several features to manipulate and organize large chunks of data.

The novel computational strategy used in *Visilab Viewer* is the result of the development of two internal modules that share a pri-

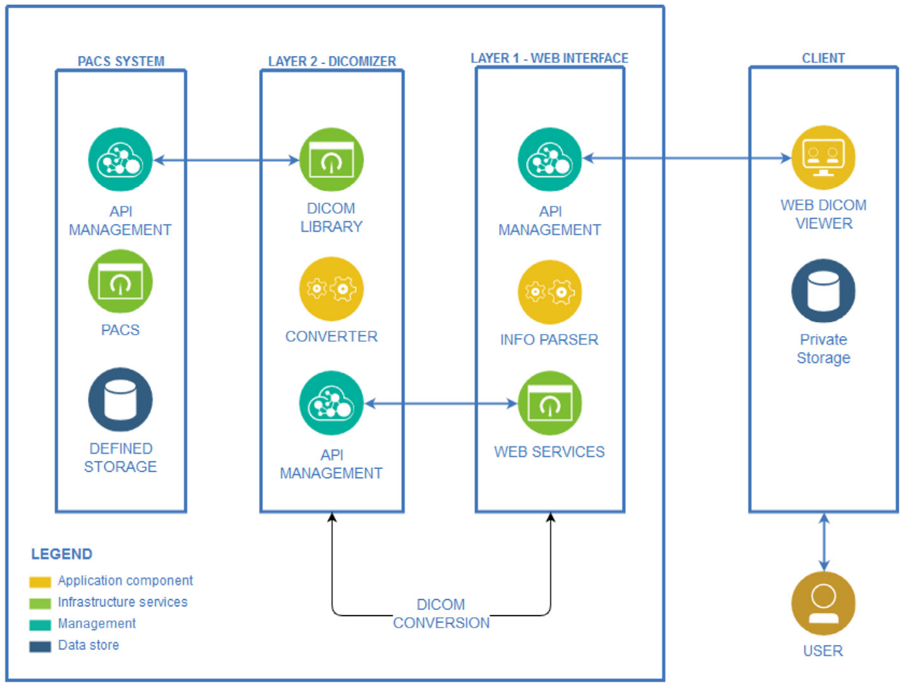


Fig. 2. Proposed architecture that supports DICOM PACS Server, conversion and visualization on the *Visilab Viewer* client.

vate storage area (see Fig. 3). The Structure Constructor module obtains the slide frame information directly from the PACS and builds a new object structure that contains the location coordinates of all frames organized inside the pyramid, by level, row and column. When this object is finally created, this first module stores it in the client storage. Secondly, the Tiling Cache Aggregator module is in charge of consulting in real time this information object in order to prepare and load in the browser cache the next promising areas that the user may observe next. In the same figure, the current user viewport corresponds to the blue area while yellow areas represent all the adjacent tiles that are likely to be visited in the same, previous and next levels. Therefore, for nearby zooming and panning events, all areas required will be already loaded in the viewport.

The client application is not just the *Visilab Viewer* but also includes a complete Laboratory Information System (LIS) where pathologists can introduce new patients, studies, series and instance objects to fill and track specimens and slides alongside the PACS. The configuration of the PACS connection related to the viewer is a key piece to handle scalability.

2.4. Digital pathology DICOM connectathon

The AIDPATH Project sponsored the first and second DICOM Connectathons ever held, where the present work was exhibited. At the first Connectathon, San Diego 2017 Pathology Visions, we provided a visualization system connected to the main data provider DICOM PACS by Pathcore, shown in Fig. 4a. During this actual industrial application, some proprietary slides were converted inside the present implementation and were inserted in the PACS.

At the second Connectathon, ECDP Helsinki 2018, we provided the complete architecture shown in Fig. 2. In this case, all participant scanners and viewers exchanged data with the VISILAB PACS (see Fig. 4b). Once again, the visualization results were satisfactory with *Visilab Viewer*.

Data generated up to this point, representative of a real industrial scenario, had been stored in the VISILAB architecture. This critical fact meant a reliable data source and results were available to be used in the following experiments.

2.5. Experiments

Once data was obtained as a result of both specific vendor format conversions and the scanning in real time and stored in the PACS, two groups of experiments were designed. The first type provided a quantitative estimation and was focused on average load latency, as this is the main factor driving the acceptance by the final user, i.e., pathologists. The second set of experiments was carried out by six pathologists to perform a qualitative assessment. The combination of quantitative (objective) and qualitative (subjective) evaluation provides a realistic view of the solution as a useful tool for digital pathology.

To validate the results obtained with *Visilab Viewer*, all slides were also tested with another three public and open source map browsers with pyramidal support: GoogleMaps [14], Leaflet [15] and IIPMooViewer [16] (hereafter IIP). Then the four web viewers are compared.

Several properties were evaluated to measure the quality of each web viewer.

- Latency: The frame load latency measures the time elapsed since the web viewer requests a frame from the server until it is displayed on the user screen.
- Panning: Panning navigation demonstrates how the web viewer responds during user navigation over the x and y axes in the image.
- Zooming: The zooming navigation is similar to pan but for the z axis.

The common objective of the panning and zooming properties is to evaluate the efficiency of the visualization of the frames in each viewer when a user is moving in any of the axes within the pyramid.

Two more features are measured from the user's point of view.

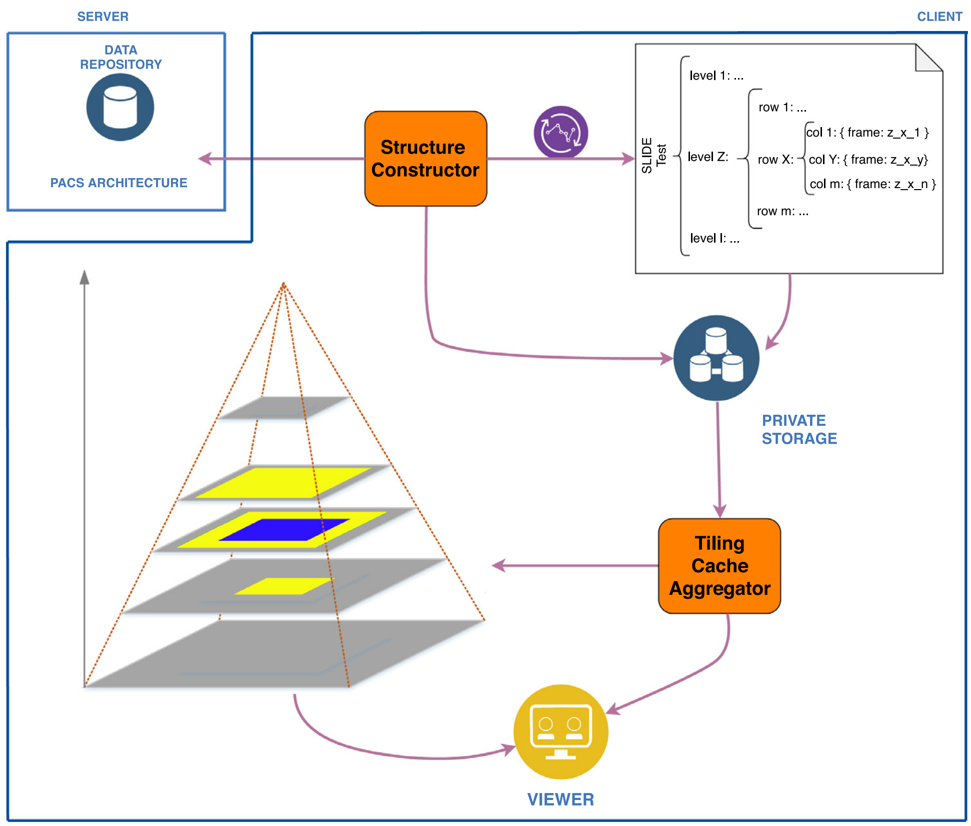


Fig. 3. *Visilab Viewer* computational method based on two key modules. The Structure Constructor calculates the organization of the total number of frames of the slide and stores a new object into the private storage. The Tiling Cache Aggregator consults the same object to load in real time the adjacent tiles (yellow) of the current user view (blue). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

- Speed: The speed during navigation shows how the user considers that the web viewer responds with an acceptable time.
- Usability: This evaluates if the interaction with the web viewer is easy and shows, in general, a suitable behavior to help in a diagnostic process.

The technical environment used to achieve the most accurate results must be the same in each of the sessions, both for quantitative and qualitative experiments. *Visilab Viewer* is able to run in most of the widely used public web-browsers (Safari, Google Chrome, Firefox, IE) and supports touch screen events. Google Chrome was selected to carry out the experimentation since this web browser provided the best support for the four viewers. The system was installed in a DS716+II Synology NAS server with OS DSM 6.2.1–23824 connected to the local intranet. Each architecture module was managed in a separate environment by the use of Docker containers. Finally, the experiments were launched in the same machine, a Dell Workstation Precision 5510 with OS Windows 10 Pro.

In order to work with the most similar network infrastructure, all experiments were run on the same physical machine connected via Ethernet to the same switch device and inside the same subnetwork where all components of the architecture were installed. Any other applications or additional processes had been previously disabled to avoid possible delays.

2.5.1. Quantitative experiments

To evaluate the viewers from a quantitative point of view the display latency (i.e., the time elapsed from the request of the resource) is calculated. This corresponds to the time elapsed from the moment of loading each of the frames that form the DICOM structure, until its visualization in the corresponding pyramidal tile position within the viewer. For that purpose, two timestamps were considered: 1^{st}) frame request time is recorded inside a defined logfile in the client environment and 2nd) frame display time within the corresponding position on the web viewer is tracked in a second logfile also located on the client side. The difference between these two times is considered as the latency for that particular frame.

2.5.2. Qualitative experiments

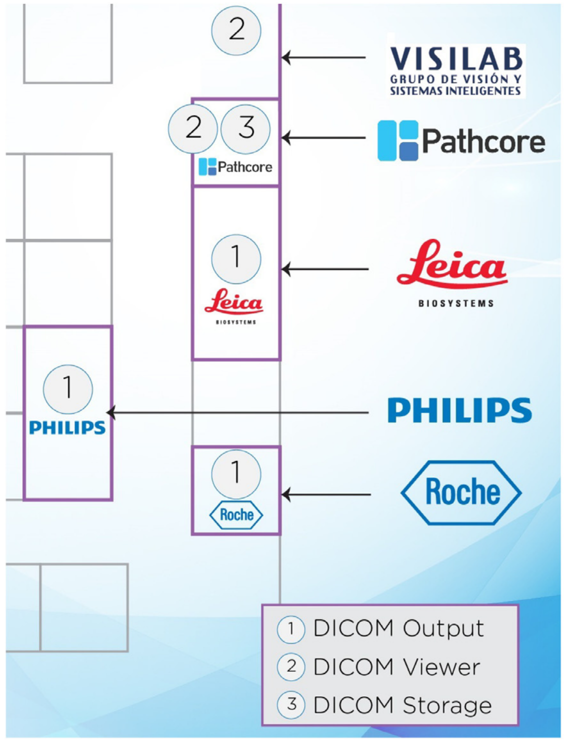
Six pathologists evaluated all WSI cases by trying the four web viewers. The viewers were anonymized to avoid influencing their perception. Once finished, each pathologist completed several questionnaires related to their user experience and the performance of the viewers.

3. Results

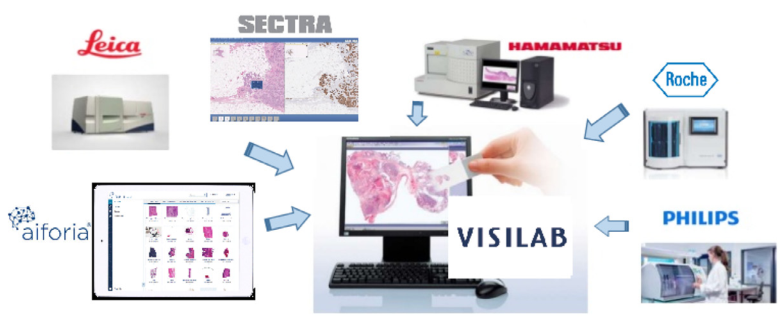
Scanned acquisition tissue slides were successfully transferred and stored in our PACS. Data was generated by the scanner vendors or by the converter installed in our system. Both types imply successful results due to the fact that *Visilab Viewer* is able to visualize them independently of the original source. Both clinicians and vendors' feedback were requested with positive comments [17].

Visilab Viewer is a web platform that contains not only the visualization engine but also a LIS environment through which physicians may create new DICOM objects from proprietary formats and upload them to the PACS.

The visualization quality of *Visilab Viewer* was validated by comparing the study of 10 human tissue slides (Table 1) with another



(a) 1^{st} DICOM Connectathon. Diagram courtesy of the WG26.



(b) 2^{nd} DICOM Connectathon event providing also storage.

Fig. 4. Vendor participants in the generation of DICOM WSI data, visualization and stored in the PACS server during the two DICOM Connectathons in digital pathology.

three public web viewers. As mentioned, the proposed web viewer is a part of the web application, which means that these three online viewers were also integrated within it. Thus, the web tool contained four viewers ready to use for the analysis of each WSI. Notice that the three public map browsers were not manipulated in any way, only configured to read the 10 WSIs as the objective is just to obtain the latency in drawing the frames.

In a first step to achieve a quantitative measurement a total number of 40 study cases was obtained from the evaluation of the 10 slides by the four different web viewers. A total number of 80 files with the initial and final timestamps was built and processed afterward to compute the average latency in each case for the four viewers.

Second, in order to provide a qualitative estimation, a total of 6 surveys were filled out by the 6 pathologists. They proceeded also to evaluate the same 40 cases (each of the 10 slides with the four web viewers separately) and then answered the questionnaire from the point of view of the user experience with each viewer.

3.1. Quantitative results

The quantitative assessment was made based on the average load time per frame, together with the number of fully loaded frames. The latter was calculated taking into account also four other parameters:

- Total Calls: Total number of frame calls launched during the start and end of the navigation process. The same frame may be requested once or several times. This value can be seen as the sum of the To Load and Cached columns.
- To Load: The actual number of frames called without repetitions. This value can be seen as the sum of the Completed and Uncompleted columns.
- Cached: Frame calls that were repeated for the same frame number and therefore not considered.
- Completed: Number of frames that a web viewer was able to load and visualize.

Table 2
Number of frames fully loaded and visualized (i.e., completed), extracted from Appendix A. Web viewers are ordered by the largest number of frames completed during navigation in each slide.

Slide	#1	#2	#3	#4
A	Visilab 271	Leaflet 233	IIP 99	GoogleMaps 94
B	Visilab 48	Leaflet 46	GoogleMaps 44	IIP 15
C	Visilab 48	GoogleMaps 45	Leaflet 35	IIP 16
D	Visilab 343	Leaflet 310	IIP 153	GoogleMaps 68
E	Visilab 256	Leaflet 212	GoogleMaps 186	IIP 140
F	Visilab 120	Leaflet 95	GoogleMaps 76	IIP 10
G	Leaflet 163	Visilab 103	IIP 89	GoogleMaps 75
H	Leaflet 217	Visilab 166	IIP 87	GoogleMaps 0
I	Visilab 305	GoogleMaps 259	Leaflet 174	IIP 121
J	Visilab 556	IIP 378	GoogleMaps 239	Leaflet 165

Table 3
Kruskal–Wallis and Dunn tests with Bonferroni correction based ordering of the web viewers. Median and IQR included for each viewer in each study case.

Slide	#1	#2	#3	#4
A	IIP 1011 (592, 2639)	GoogleMaps 1886 (1167, 4582.5)	Visilab 2473.5 (1440, 4046.25)	Leaflet 3801 (1329.25, 10709.75)
B	Visilab 693 (565, 1507.5)	GoogleMaps 913 (793, 1203.5)	IIP 987 (721.75, 4444.75)	Leaflet 1812.5 (911.5, 2778.75)
C	Visilab 549 (317, 879.5)	IIP 851 (850, 1187.5)	GoogleMaps 1097 (780.5, 1444.25)	Leaflet 5904 (1634, 9381)
D	IIP 1203 (799.75, 1851.75)	GoogleMaps 2082 (1199, 4453)	Visilab 2437 (1640, 4806.25)	Leaflet 3632 (1541, 7267)
E	IIP 809 (687.5, 1001)	Leaflet 945.5 (591.75, 1232)	GoogleMaps 1270 (1177, 1364)	Visilab 1430 (731.25, 2037)
F	Visilab 628 (357.5, 684)	IIP 1095 (577.5, 1096)	GoogleMaps 2072 (1550.25, 2383)	Leaflet 2580 (1847, 4105)
G	Visilab 1213 (1058, 1725)	IIP 1701 (1610, 3200)	GoogleMaps 1701 (1291, 2497.5)	Leaflet 3259 (1708, 11607)
H	IIP 1695.5 (1176.25, 2591)	Visilab 1967 (1225, 2655)	Leaflet 4121.5 (2177.25, 6026.5)	
I	Visilab 1107 (806, 1375)	IIP 1159 (798, 1851)	Leaflet 2645 (1595, 3800)	GoogleMaps 3573.5 (2251.25, 5476)
J	IIP 1629 (1081, 2883)	Leaflet 2314.5 (1405, 6524.25)	GoogleMaps 2569 (1995, 5326)	Visilab 2916 (1890, 4713)

Table 4
Final ranking of the four web viewers. The selection is based on the positions obtained in Tables 3 and 2. Points are calculated by multiplying the number of occurrences in each table. The web viewer with the highest value is selected as the most suitable in the ranking. For example, in the case of *VisilabViewer*, it achieved the #1 position, since it appears 8 times in Tables 3 and 5 times in Table 2, 40 points in total. In the next score *VisilabViewer* is not considered. The ranking ends with IIP in the last position.

Rank		Selected	Navigation time (ms)
#1	Visilab: 40 GoogleMaps: 0 Leaflet: 0 IIP: 0	Visilab	147868.6 (2'28")
#2	GoogleMaps: 6 Leaflet: 10 IIP: 4	Leaflet	131772.1 (2'12")
#3	GoogleMaps: 20 IIP: 4	GoogleMaps	107324.7 (1'47")
#4		IIP	108359.3 (1'49")

- Uncompleted: Number of frames that were requested but were never loaded and therefore never visualized.

All these parameters values are shown in Appendix A, where the total number of calls per slide and the results achieved by each of the four web viewers are displayed. Table 2 summarizes the results shown in Appendix A, where the web viewers are sorted by the highest number of fully loaded and visualized frames, i.e., completed frames. *Visilab Viewer* performed the highest number of completed frames in 8 of the 10 WSI cases. LeafLet was the second best viewer. On the other hand, IIP completed the lowest number of frames but GoogleMaps got the worst score because it was not able to load and visualize one of the WSI, image H.

The analysis of the viewers' behavior was performed next, based on the average load time per frame in each case. We assume as a null hypothesis, *Ho*, that there are no significant differences in the average load response between each of the viewers. Later the alternative hypothesis is accepted, and in addition a second contrast was done to generate a ranking of the web viewers. The lowest average value obtained is therefore considered to represent the optimal web application.

Shapiro–Wilk non-parametric tests was applied to checked whether the distributions of all the recorded time measures follow a normal distribution for each of the cases. Then, the Kruskal–Wallis contrast was used to study the best average performance in terms of latency and robustness.

The boxplots of the data sets in which the Kruskal–Wallis test was applied are depicted in Appendix B for each slide or study case. All test results were significant with a 95% of confidence, so *Ho* was rejected, and the conclusion was that in all cases at least one web viewer achieved a significant different median value from the other web viewers.

The Dunn test [18] together with Kruskal–Wallis were applied in all cases to rank web viewers considering the median and the interquartile range (IQR) of the frame latency. These values are shown in Table 3. The results of this table represent the ordering from the lowest to the highest median value corresponding to each of the boxplots in Appendix B.

Five out of ten cases ranked *Visilab Viewer* with the lowest median, i.e., it loads and displays the frames of those 5 slides slower than the other three web viewers. However, the ranking may be controversial if considered independently from Table 2. The relation between these two results is important as it explains that the more frames loaded the higher the bias in the median value achieved and boxplots shape will tend to enlarge.

The total number of frames inspected is strongly relevant. Therefore, the total number of frames that each viewer was able to load in its viewport is critical. The Kruskal–Wallis test should be accompanied with Table 2. Taking into account both rankings, a score based on the number of occurrences in each of them has been calculated to conclude the final selection. This is shown in Table 4.

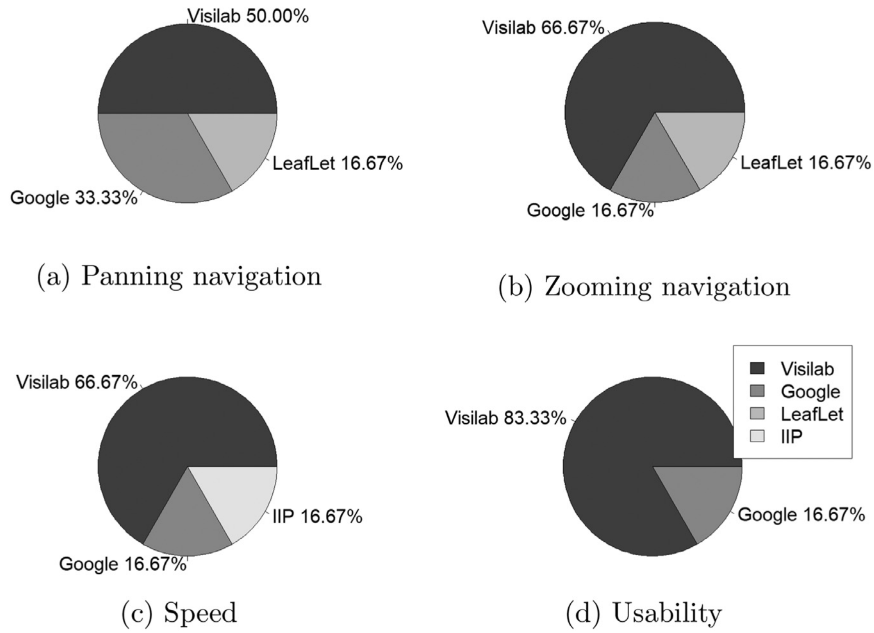


Fig. 5. Results from the six pathologist surveys after evaluating each web viewer. The diagrams represent the summary preference of all variables considered in the evaluation.

Results also confirmed that the intended heavy processing behavior expected in the web viewers happened. In the case of slide H, for example, GoogleMaps was not able to render any data.

The query calls for each web viewer were inspected. IIP showed a high number of calls because every time that the user makes a panning move the entire viewport is updated and therefore the frames are requested again. Therefore the number of cached calls is also high.

In the bloxplots, it is observed that IIP IQR is, in general, the shortest while Leaflet obtained the opposite, i.e. the largest IQR with still many outliers in some cases. *Visilab Viewer* achieved the highest number of loaded frames but also shorter IQR in its boxplots; hence, it may be considered the most robust application.

3.2. Qualitative results

Each pathologist evaluated, individually and in separate sessions, the 10 slides with the 4 web viewers. The main features evaluated were the panning and zooming during navigation. Finally, they were asked which of the four web viewers they would select as the best tool for daily use.

Results are shown in Fig. 5. *Visilab Viewer* obtained the highest usability percentage, 83.33%, as a potential web software to be used in the future for diagnosis. Panning navigation was more distributed and zooming was also preferred with *Visilab Viewer*. IIP obtained nil values in usability and zooming navigation. Leaflet also obtained nil values in the case of the speed feature. The average navigation time for the 10 slides is also indicated as part of the summary (last column of Table 4).

Pathologists also indicated the reasons why some slides could not be evaluated with some viewers. The main drawbacks were: a) GoogleMaps and Leaflet do not support large magnifications and the image disappears when zooming. The highest magnification is essential in some cases to perform a correct diagnosis, for example, WSI case E of skin tissue; b) Leaflet deforms large images, as it is the case of images H and J; c) Usually the blinking effect when zooming was the most disturbing effect especially when navigating with IIP; d) most of the viewers are slow displaying the frames; and e) GoogleMaps could not open slide I in any session and IIP could not open it in two of the six sessions.

They pointed out that the navigation with *Visilab Viewer* resulted without blinks and was the most uniform, being able to open all WSIs in all sessions.

4. Discussion

Connectathons are essentials to measure how hardware and software solutions from different scanner manufacturers can successfully exchange data following the DICOM standard and messaging protocol. They allow testing interoperability and compatibility. The performance of the *Visilab Viewer* was comparable with other proprietary desktop viewers. This fact was expected because the raw data persists in DICOM format inside the provided PACS. The architecture solution proposed in the present work was successfully tested, which, in comparison to other solutions, also includes several modules with additional functionalities together with the scalable web viewer.

Despite the many proposals and vendor solutions that work properly in their proprietary architectures, they are not proved completely in terms of actual compatibility. Many vendors define their DICOM solutions but with the lack of actual testing with other hardware and software providers. Some existing architectures are functional but in a standalone environment that implements the standard strictly, as explained in [11]. Final users will work ignoring if their data is factually compatible with other DICOM users and viceversa. In other words, the use of this standard is possible, but that does not imply that a commercial product available today will be able to manage large production data sets [12]. There is a practical and critical difference to test in each vendors product between exporting single slides and transmitting them. At this point, Connectathons are the actual interoperability tests and are therefore required.

There is a constant effort to improve and add new features to the standard to contribute a more robust solution for the digital pathology area. WG26 addresses issues during its meeting and it also establishes the new challenges to solve when a new requirement is identified during the Connectathons. These issue case proposals are listed in [19].

The second objective of this work was to validate *Visilab Viewer* compared to other existing solutions. Apart from being selected as the first choice to use during diagnosis by the majority of pathologists, we noted that their main preference was based on soft navigation and high WSI quality visualization. The soft feature was mainly referred to as a rapid visualization of frames without any blinks or inconsistencies while panning or zooming. In other web viewers, these inconsistencies consisted, for example, in showing the frames in incorrect positions, reducing the dimensions of the image or even not being able to visualize the slide.

An important point to note is that the three solutions selected for the experiments are not specialized in the visualization of digital pathology slides. *Visilab Viewer* was not originally conceived for digital pathology, albeit the development carried out has focused on this type of images. All of the viewers share the basic functional goal of displaying extremely large images in the most efficient way possible.

From the quantitative results *Visilab Viewer* achieved the best average latency in 50% of cases and IIP the other 50%. This fact may be misleading because *Visilab Viewer* was also the web viewer able to display the highest number of frames fully loaded. On the other hand, IIP was ranked in the last position with the fewest number of loaded frames. This is the reason why the average latencies differ. While this last viewer executes many queries at first, in most cases they are repeated calls to finally be able to display only some of them. From the point of view of the final user, they preferred the behavior of *Visilab Viewer* in terms of zooming, panning ad speed. Finally, Leaflet was ranked in the last position in most of the experiments in terms of latency, although it was also capable of loading a high number of frames.

Furthermore, the pyramid structure requires more elements while zooming deeper and all viewers experienced an exponential loading time while navigating through deeper levels in very fast navigation. We considered to improve the web service to make it faster to send the frames and pre-build also the pyramidal structure with the relation of each of the frames and its coordinates in the middleware layer instead of the web platform. In parallel, we considered the rapid accumulation of cached data inside any web browser. Some of the WSI sizes implied over 4GB of data that the browser must handle in case of full navigation. The lifetime of resources in the cache memory is set by default to seconds or even days and pathologists should not be required to manipulate these settings. Browsers' garbage collectors may also slow down the loading of frames in some seconds.

4.1. Future work

Connectathons allow devices and medical record companies to test and perfect their interoperability capabilities in week-long collaborative environments around the world [20]. The purpose of these connectivity tests is to fill the gaps in any technical field, both software and hardware. Alongside with the promotion of the use of DICOM, these two goals will complete the interoperability cycle for exchanging data, tags metadata and image analysis. The cooperation to promote successive Connectathons is a need to maintain and improve these objectives.

The information and DICOM tags integration from different clinical areas is required, especially, in cancer diagnosis where correlation and integration of radiology and pathology are not properly covered yet [21]. Uncoordinated communication of these two clinical areas triggers discordance that requires additional time to resolve the cases. To solve this problem, all clinical data may be integrated into the same web platform.

Algorithms based on artificial intelligence techniques will help during diagnosis as extra tools to identify in real time areas of interest or any other data defined by pathologists [22]. These methods will be integrated as tools to run from the web platform and inside *Visilab Viewer*. Other user tools are the inclusion of annotations for educational purposes or to help by comments during collaborative diagnosis.

5. Conclusions

A complete novel architecture was developed to support compatibility and interoperability among tissue pathology scanners, aiming to visualize the acquisitions based on the DICOM standard. This is a reliable solution because it was tested in current industrial experiments during two worldwide Connectathons organized by the DICOM WG26. Therefore, we provided a vendor-neutral archive (VNA) with successful results for visualization of whole slide images in Anatomical Pathology.

A web platform was also provided, including an online pyramid viewer named *Visilab Viewer* able to show all the DICOM formats generated during the connectivity tests and via a converter tool also developed by the authors. This platform is scalable and able to integrate different PACS connections where *Visilab Viewer* can require the data to visualize by pathologists. At the same time, users can generate their DICOM object over a LIS interface and also send, from the same web, the scanned tissues to the PACS. The client can compute the organization of the entire pyramid per level, i.e., each frame x and y coordinates. This structural information is stored in a private database to be consulted afterward in real time by an aggregator module. This module pre-loads in the cache browser the adjacent tiles of the current user viewport.

The validation method consisted of including three other public web pyramid browsers to evaluate a set of ten WSIs. The particular slides were selected to generate the highest processing requirements by each of the viewers. The average latency in displaying the frames of each slide was compared. *Visilab Viewer* loaded the highest number of frames in most of the cases and was ranked between the first and second position in latency average.

Six independent pathologists participated in the testing phase of the four web viewers with a set of double-blind experiments. They tried to generate the diagnosis of all the DICOM slides by using each of the four web viewers. They focused on the zooming and panning response to finally complete a survey. The results demonstrated that they preferred *Visilab Viewer* in the first position in panning, zooming and usability, and therefore as the best one for their daily diagnosis work.

Author declaration

We wish to confirm that there are no known conflicts of interest associated with this publication entitled "Optimum Web Viewer Application for DICOM Whole Slide Image Visualization in Anatomical Pathology". Sincerely, Nieves Lajara.

Acknowledgments

The authors acknowledge financial support of the European Union FP7 programme under grant agreement No. 612471 AIDPATH (http://aidpath.eu/). They also express their gratitude to the Anatomic Pathology Service from the Ciudad Real Hospital for the physicians' anonymous participation during the evaluation tests of the work presented.

Appendix A. Frame Calls Summary

The total number of calls per slide and the results achieved by each of the web viewers are shown. Highlighted rows correspond to the web viewer that completed in its pyramidal grid the highest number of frames. Numbers in bold represent the viewers that requested the highest actual number of frames expected to visualize (Tables A1–A10).

Table A1
Calls summary from Slide A.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	375	**300**	75	**271**	29
GoogleMaps	191	94	97	94	0
LeafLet	410	257	153	233	24
IIP	226	99	127	99	0

Table A2
Calls summary from Slide B.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	66	**48**	18	**48**	0
GoogleMaps	119	44	75	44	0
LeafLet	243	46	197	46	0
IIP	84	15	69	15	0

Table A3
Calls summary from Slide C.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	66	**48**	18	**48**	0
GoogleMaps	85	45	40	45	0
LeafLet	297	48	249	35	13
IIP	77	16	61	16	0

Table A4
Calls summary from Slide D.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	586	**456**	130	**343**	113
GoogleMaps	178	68	110	68	0
LeafLet	437	339	98	310	29
IIP	239	153	86	153	0

Table A5
Calls summary from Slide E.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	343	**256**	87	**256**	0
GoogleMaps	307	186	121	186	0
LeafLet	634	212	422	212	0
IIP	318	140	178	140	0

Table A6
Calls summary from Slide F.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	169	**120**	49	**120**	0
GoogleMaps	98	76	22	76	0
LeafLet	141	103	38	95	8
IIP	73	10	63	10	0

Table A7
Calls summary from Slide G.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	142	111	31	103	8
GoogleMaps	125	75	50	75	0
LeafLet	412	**163**	249	**163**	0
IIP	151	89	62	89	0

Table A8
Calls summary from Slide H.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	202	172	30	166	6
GoogleMaps	–	–	–	–	–
LeafLet	253	**219**	34	**217**	2
IIP	146	87	59	87	0

Table A9
Calls summary from Slide I.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	390	**325**	65	**305**	20
GoogleMaps	361	259	102	259	0
LeafLet	197	181	16	174	7
IIP	238	121	117	121	0

Table A10
Calls summary from Slide J.

	Total Calls	To Load	Cached	Completed	Uncompleted
Visilab	695	**600**	95	**556**	44
GoogleMaps	362	239	123	239	0
LeafLet	177	165	12	165	0
IIP	521	378	143	378	0

Appendix B. Statistics Results

The user experiments will potentially differ from each other. A particular user will navigate through some areas that will differ from the areas navigated by other individual. That is, different spaces from the pyramid will be requested and therefore, the tracking of the same frames is not possible in every case. Therefore variables are unpaired so it is not possible to opt for the use of the Friedman test. Consequently we considered the Kruskal–Wallis contrast. as the most suitable non-parametric test to study the best average performance in terms of latency and robustness based on the median value analysis (Figs. B6–B15).

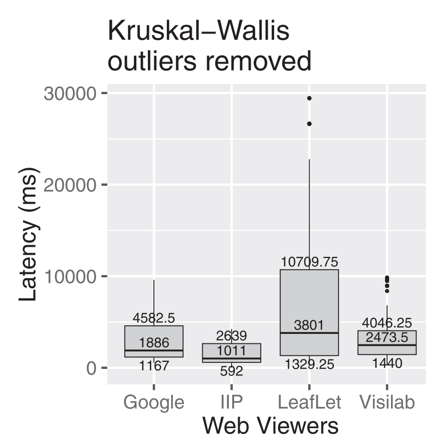


Fig. B1. Slide A Kruskal–Wallis contrast without outliers.

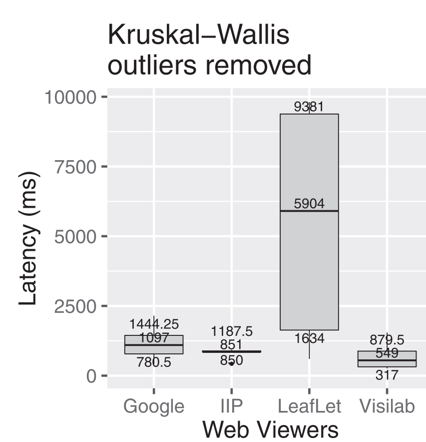


Fig. B2. Slide C Kruskal–Wallis contrast without outliers.

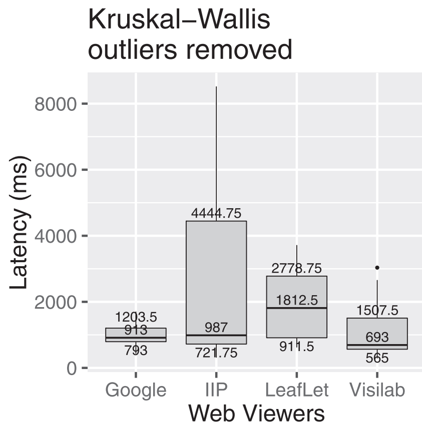


Fig. B3. Slide B Kruskal–Wallis contrast without outliers.

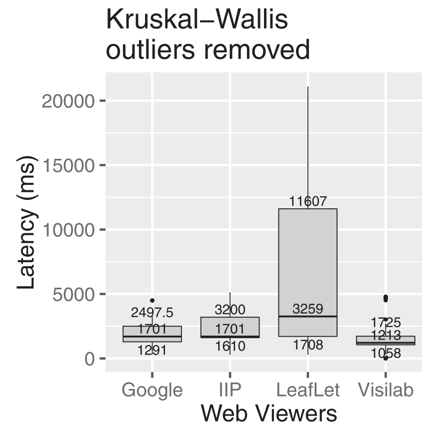


Fig. B6. Slide G Kruskal–Wallis contrast without outliers.

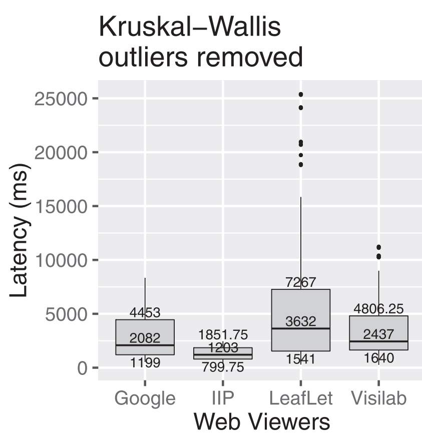


Fig. B4. Slide D Kruskal–Wallis contrast without outliers.

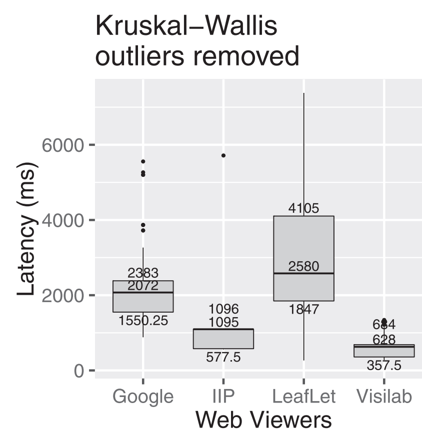


Fig. B7. Slide F Kruskal–Wallis contrast without outliers.

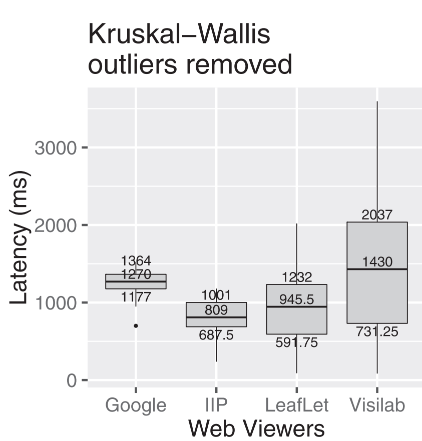


Fig. B5. Slide E Kruskal–Wallis contrast without outliers.

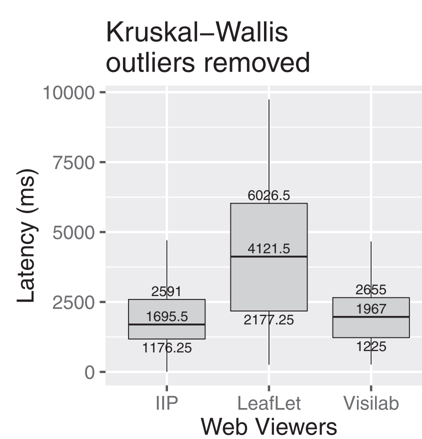


Fig. B8. Slide H Kruskal–Wallis contrast without outliers.

The boxplots of the data sets in which the Kruskal–Wallis test was applied are shown here for each slide or study case. The values of these boxplots and medians agree with the Kruskal–Wallis contrast values. At the beginning of the study the boxplots contained too many outliers. If included, they could bias results. Therefore these first outliers were removed in order to clean the noise and avoid aberrant values in the statistical tests.

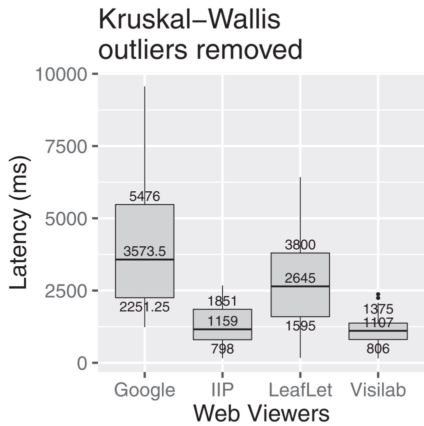


Fig. B9. Slide I Kruskal–Wallis contrast without outliers.

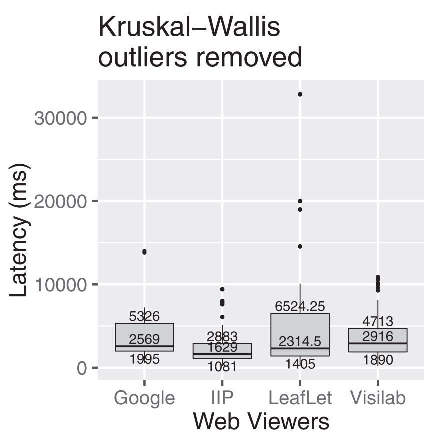


Fig. B10. Slide J Kruskal–Wallis contrast without outliers.

References

[1] N. Farahani, M. Riben, A. Evans, L. Pantanowitz, International telepathology: promises and pitfalls, Pathobiology 83 (2–3) (2016) 121–126, doi:10.1159/000442390.

[2] N. Farahani, L. Pantanowitz, Overview of telepathology, Surg. Pathol. Clin. 8 (2) (2015) 223–231, doi:10.1016/j.path.2015.02.018.

[3] S. Thorstenson, J. Molin, C. Lundström, Implementation of large-scale routine diagnostics using whole slide imaging in sweden: digital pathology experiences 2006–2013, J. Pathol. Inf. 5 (14) (2014), doi:10.4103/2153-3539.129452.

[4] A.C.R. Nema, Digital imaging and communications in medicine (DICOM) Supplement 122: Specimen Identification and Revised Pathology, 2008, (Available at ftp://medical.nema.org/medical/dicom/final/sup122_ft2.pdf (Accessed 2018-12-17)).

[5] A.C.R. Nema, Digital imaging and communications in medicine (DICOM) Supplement 145: Whole Slide Imaging in Pathology, 2010, (Available at ftp://medical.nema.org/medical/dicom/final/sup145_ft.pdf (Accessed 2018-12-17)).

[6] N. Farahani, A.V. Parwani, L. Pantanowitz, Whole slide imaging in pathology: advantages, limitations, and emerging perspectives, Pathol. Lab. Med. Int. 7 (2015) 23–33, doi:10.2147/PLMI.S59826.

[7] G. Bueno, M.M. Fernández-Carrobles, O. Deniz, M. García-Rojo, New trends of emerging technologies in digital pathology, Pathobiology 83 (2–3) (2016) 61–69, doi:10.1159/000443482.

[8] A. Parwani, L. Hassell, E. Glassy, L. Pantanowitz, Regulatory barriers surrounding the use of whole slide imaging in the United States of America, J. Pathol. Inf. 5 (1) (2014) 38, doi:10.4103/2153-3539.143325.

[9] M.G. Hanna, L. Pantanowitz, A.J. Evans, Overview of contemporary guidelines in digital pathology: what is available in 2015 and what still needs to be addressed? J. Clin. Pathol. 68 (7) (2015) 499–505, doi:10.1136/jclinpath-2015-202914.

[10] G. Siegel, D. Regelman, R. Maronpot, et al., Utilizing novel telepathology system in preclinical studies and peer review., J. Toxicol. Pathol. 31 (4) (2018) 315–319, doi:10.1293/tox.2018-0032.

[11] T.M. Godinho, R. Lebre, L.B. Silva, C. Costa, An efficient architecture to support digital pathology in standard medical imaging repositories, J. Biomed. Inf. 71 (2017) 190–197, doi:10.1016/j.jbi.2017.06.009.

[12] D. Clunie, D. Hosseinzadeh, M. Wintell, et al., Digital imaging and communications in medicine whole slide imaging connectathon at digital pathology association pathology visions 2017, J. Pathol. Inf. 9 (6) (2018), doi:10.4103/jpi.jpi_1_18.

[13] I. Gilman, OpenSeaDragon, An open-source, web-based viewer for high-resolution zoomable images, 2017, (Available at https://openseadragon.github.io (Accessed 2018-12-17)).

[14] Google Inc, Google Maps JavaScript API V3 Reference, 2017, (Available at https://developers.google.com/maps/documentation/javascript/reference/ (Accessed 2018-12-17)).

[15] V. Agafonkin, Leaflet API reference 1.3.4, JavaScript library for mobile-friendly interactive maps, 2017, (Available at https://github.com/Leaflet/Leaflet (Accessed 2018-12-17)).

[16] E. Bertin, R. Pillay, C. Marmo, Web-based visualization of very large scientific astronomy imagery, Astron. Comput. 10 (2015) 43–53, doi:10.1016/j.ascom.2014.12.006.

[17] D. Hosseinzadeh, A Multi-Vendor Demonstration of DICOM Workflow for Pathology, 2017, (Available at http://www.ftp://medical.nema.org/medical/Dicom/DICOMWSI/Connectathon/PV2017/documents/DICOM%20Digital%20Pathology%20Connectathon%20Proposal.pdf (Accessed 2018-12-17)).

[18] O.J. Dunn, Multiple comparisons using rank sums, Technometrics 6 (3) (1964) 241–252, doi:10.1080/00401706.1964.10490181.

[19] DICOM working group WG26, DICOM Case and Correction Proposals, 2018, (Available at ftp://medical.nema.org/../MEDICAL/Dicom/CP/ (Accessed 2018-12-17)).

[20] E.B. Sloane, A. Thalassinidis, R.J. Silva, ISO/IEEE 11073, IHE, and HL7: Fostering Standards-Based Safe, Reliable, Secure and Interoperable Biomedical Technologies (2018) 1–3. doi:10.1109/SECON.2018.8479122.

[21] C.W. Arnold, W.D. Wallace, S. Chen, et al., RadPath: A Web-based system for integrating and correlating radiology and pathology findings during cancer diagnosis, Acad. Radiol. 23 (1) (2016) 90–100, doi:10.1016/j.acra.2015.09.009.

[22] A. Madabhushi, G. Lee, Image analysis and machine learning in digital pathology: challenges and opportunities, Med. Image Anal. 33 (2016) 170–175, doi:10.1016/j.media.2016.06.037.