Streaming of medical images using JPEG2000 interactive protocol

Jean-François Pambrun and Rita Noumeir*
Department of Electrical Engineering,
École de Technologie Supérieure, University of Quebec
1100 Notre-Dame West, Montreal, Quebec, Canada, H3C 1K3
E-mail: jean-francois.pambrun.1@ens.etsmtl.ca
E-mail: rita.noumeir@etsmtl.ca
*Corresponding author

Abstract: Access to all relevant information at the diagnostic decision moment improves the quality of care. With the deployment of the Electronic Health Records (EHR), information now exits in multiple distributed systems. When conducting diagnosis based on medical images, physicians often need to compare current images with older ones that may need to be imported prior to visualization. This leads to persistency management and information consistency issues. JPEG2000 Interactive Protocol (JPIP) can be used to avoid imports by streaming images and significantly reduce bandwidth requirements. In this paper, we describe how JPIP can enable two interactive visualization schemes, "zoom & pan" and a magnification lens, to significantly improve user experience and productivity while minimizing network usage. We will show that, with carefully selected compression parameters, JPIP allows physicians to start their investigation within few seconds of the initial request and stay productive during the entire download process.

Keywords: Medical Imaging, Electronic Health Record, Image Communication, Image Streaming, JPIP, JPEG2000

Biographical notes: Jean-Francois Pambrun is currently working toward his Ph.D from the department of electrical engineering at the École de Technologie Supérieure(ETS), Université du Québec where he also received his Bachelor's degree in electrical engineering in 2009. He is the main author of the IHE testing software for the Patient Identification Cross-Reference (PIX) and Patient Demographics Query (PDQ) using Health Level 7 (HL7) version 3 (v3). His research interests include medical image streaming and compression, healthcare informatics and distributed system interoperability.

Rita Noumeir is a professor at the Department of Electrical Engineering, at the École de Technologie Superieure, University of Quebec in Montreal. Her main research interest is in the Healthcare Information Technology, specifically, Interoperability, Electronic Patient Record, Security, Information Confidentiality, Image Processing and Information Visualization. Dr. Noumeir holds a Ph.D. and Master's degrees in Biomedical Engineering from École Polytechnique, University of Montreal. She has published and lectured extensively in biomedical informatics, software analysis and image processing. She has also provided consulting services in architecture analysis, workflow analysis, technology assessment and image processing for several software and medical companies including Canada Health Infoway.

1 INTRODUCTION

The Electronic Health Record (EHR) enables informed health decision by making all relevant prior diagnostic information such as observations, laboratory results and images instantly available from anywhere. That is to say, independently from the geographic locations of the point of access and the institution where the information was initially gathered. The deployment of EHR is expected to improve the quality of care by enabling more informed decisions; it is also expected to improve the efficacy and efficiency of the overall healthcare system by improving productivity and by reducing the duplication of information gathering. Canadian authorities are currently in the process of implementing a pan-Canadian EHR and expects(**?**) improved digital imaging report turnaround time of 40% with subsequent treatments starting 10-24 hours sooner on average thus significantly reducing length of stays. They also anticipate improved referring physician efficiency of 50 minutes per week enabling annual savings of 160M$ as well as a 2-3% reduction of exam duplication (from 3-4% overall; 1.7% within institutions). These duplicates represent about 1.3 million exams each year in Canada, costing about 71M$ while representing health hazards to patients.

EHR implementation is very challenging because they are not monolithic systems that can be provided by a single manufacturer. They are virtual systems that result from the cooperation of several heterogeneous distributed systems providing ubiquitous access to the diverse diagnostic information related to a specific patient. Interoperability is therefore essential. Achieving interoperability requires the use of communication standards; it also requires common vocabularies, common semantics, process flows that are agreed on, as well as encoding and semantics mapping between different standards. Interoperability in healthcare has been very difficult to achieve; it is costly and frequently requires specific integration interfaces despite the existence of medical standards for many decades now. Even though, standards are necessary, alone they are not sufficient. They enable interoperability within a limited scope, for a specific clinical domain or a specific function. To close this gap, Integrating the Healthcare Enterprise (IHE) provides a process for building a detailed framework for the implementation of standards. It started in 1998 and was sponsored jointly by the Radiological Society of North America (RSNA) and the Healthcare Information and Management Systems Society (HIMSS). Currently, several other associations sponsor IHE. It has expanded over several clinical domains and benefits from broad international support. IHE follows a process where the care providers identify the needs for integration and where the manufacturers and information technology professionals cooperate and agree upon an implementation of established standards to provide a solution for each identified problem(**?**). IHE recently defined an architectural infrastructure for enabling document sharing between multiple enterprises. This is known as the Cross-Enterprise Document Sharing Integration Profile (XDS). XDS lays the basic framework for deploying regional and national EHR by addressing the needs for the registration, distribution and access across health enterprises of patient's documents. Its architecture is based on a central registry that holds metadata information describing published documents residing in one or multiple document repositories. As medical images constitute important information of the patient health record, XDS has been extended to include images. As the result of an extensive investigation effort of several design solutions(**?**), the Cross Enterprise Document Sharing for imaging (XDS-I) was published as part of the IHE Technical Framework.

The deployment of XDS-I as the framework for sharing images within the EHR is taking place in many countries including Canada, USA, Japan and several European countries. As part of those deployment efforts, several difficulties have emerged. Some of these challenges are related to information distribution and scalability (**?**) but the main origin of these difficulties resides in the need to compare old images with current ones, as part of the diagnostic interpretation process of current images. In fact, to conduct the interpretation, the radiologist usually compares the current images with prior ones that may have been acquired in a different enterprise. With the EHR, the radiologist knows about the existence of those priors and can access them. However, comparison is conducted within a single software application that offers specific operations for medical imaging interpretation, such as a synchronized navigation between two different image sets. This application is thus required to have access to both image sets.

Presently, most medical imaging applications assume images are under their complete control. In other words, they assume all images are identified and managed in a single consistent way. This assumption does not hold when foreign images need to be imported into the system. In fact, identification schemes are different between several enterprises and may result in identification that is not unique. Patient and order identifications are such examples.

In addition, importing foreign images into a local application creates a new major problem related to persistency management and information consistency. Image import is essentially image duplication. How can foreign images be identified as such so they can be deleted or discarded at the end of the process? Moreover, how to propagate corrected information to the duplicated instance?

One possible solution for all the previously stated problems lies in avoiding image import altogether. This is achievable with image streaming. Additionally, image streaming can significantly reduce bandwidth requirements when viewing large images or large image sets, by only streaming the data necessary to fulfill the user's task at the best screen resolution. A 28 mega-pixel digital mammogram, for instance, weighs about 50 megabytes, but a typical monitor can only display 2 mega-pixels making most of this data useless at screen resolution. On the other hand,

transferring a complete Computed Tomography (CT) series may require over a gigabyte but only small subset actually contains diagnostically significant information.

In this paper, we will describe how JPEG2000 Interactive Protocol (JPIP) can be used in the context of EHR to enable streaming of medical images directly from imaging sources to image processing workstations. We will first present an overview of the technologies involved in diagnostic imaging along with a standard compliant way of implementing JPIP in EHR. Two interactive visualization schemes that significantly improve bandwidth efficiency, user experience and productivity will then be presented and thoroughly evaluated. We will also discuss the effects of compression parameters on streaming performances and identify optimal configurations for both visualization schemes.

2 TECHNOLOGY OVERVIEW

2.1 Image compression with JPEG2000

JPEG is probably the most widely used image compression standard. It is used in all digital cameras and it is currently the preferred image format for transmission over the Internet. However, JPEG was published in 1992 and modern applications such as digital libraries and medical imaging now show some of its shortcomings(**??**). These deficiencies include poor lossless compression performances, inadequate scalability and significant blocking artifacts at low bit rates.
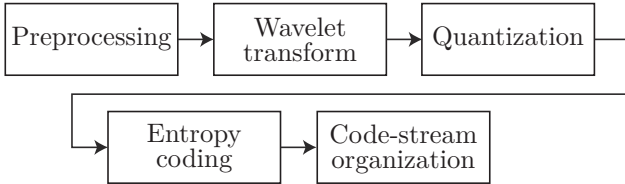

Figure 1: JPEG2000 coder block diagram

In the early 90s, researchers began working on compression schemes based on wavelet transforms pioneered by Daubechies (**?**) and Mallat (**?**) with their work on orthogonal wavelets and multi-resolution analysis. These novel techniques were able to overcome most of JPEG's weaknesses. Later, in the mid-90s, the Joint Photographic Experts Group started standardization efforts based on wavelet compression that culminated with the publication of the JPEG2000 image coding system by the International Standardization Organization (ISO) as ISO/IEC 15444-1:2000 and the International Telecommunication Union (ITU) as T.800 (**?**). Major improvements were achieved by the use of the Discrete Wavelet Transform (DWT), a departure from the Discrete Cosine Transform (DCT) used in JPEG that enabled spatial localization, flexible quantization and entropy coding as well as better stream organi-

zation. It is those enhancement that enabled new features for the JPEG2000 codec, including improved compression efficiency, multi-resolution scaling, lossy and lossless compression based on a single code-stream, Regions Of Interest (ROI) coding, random spatial access and progressive quality decoding. Most compression algorithms can be broken up in four fundamental steps(Fig. 1): preprocessing, transform, quantization, entropy coding. With JPEG2000, a fifth step, code-stream organization, enables some of the most advanced feature of the codec (random access and progressive decoding). The entire coding process is explained in the following subsections.

2.1.1 Preprocessing

JPEG2000's preprocessing involves three tasks: tilling, DC level Shifting and colour transforms. Tilling is used to split the image in rectangular tiles of identical size that are be independently encoded and may use different compression parameters. Tiles can be as large as the whole image (i.e. only one tile). They are used to reduce computational and memory requirements of the compression process. They are not typically used in diagnostic imaging because discontinuities along adjacent tiles edges tend to produce visible artifacts. Unsigned pixels values are then shifted by $-2^{(n-1)}$ so their values are evenly distributed around zero thus eliminating possible overflows and reducing the arithmetic coder's complexity. This, however, does not affect compression performance. As for colour transforms, JPEG2000 supports as many as 2^{14} components (colour channels). When pixels are represented in the RGB (Red, Green and Blue) colour space, they can be converted to luminance and chrominance channels to take advantage of channel decorrelation and increase compression performance. Two colour transforms are included in the base standard: RGB to YCbCr, called irreversible colour transform (ICT) and an integer-to-integer version, RGB to YDbDr, for reversible colour transform (RCT). The former is unsuitable for lossless coding because of rounding errors caused by floating point arithmetic. Both DC level shift and colour transform are reversed at the decoder.

2.1.2 Transform

As mentioned earlier, the Discrete Wavelet Transform (DWT) is at the core of JPEG2000's implementation. Uni-dimensional forward DWT involves filtering the input signal by a set of low and high pass filters that are referred as analysis filter bank. Filtering with the analysis bank produces two output signals that, once concatenated, are twice as long as the input. They are then subsampled by dropping every odd coefficient reducing the number of samples to the same amount that was present in the original signal (plus one for odd length input signals). Those wavelet filter taps are especially selected to allow perfect reconstruction regardless of this sub-sampling. The result is a smaller blurred version of the original signal along with its high frequency information. The process can be
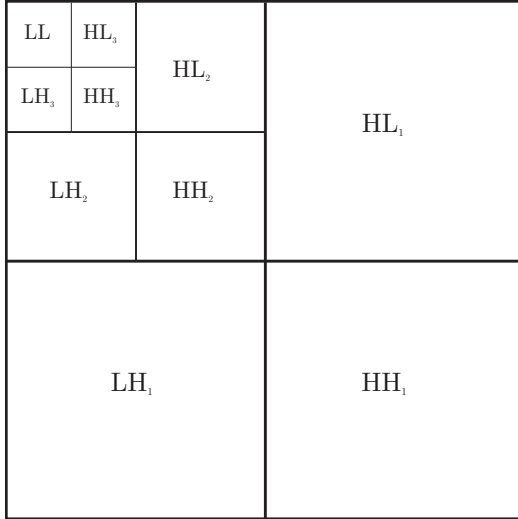
Figure 2: Wavelet subbands for a three levels decomposition

reversed by applying the inverse synthesis filter bank; coefficients are up-sampled by inserting zeros between every other coefficients and the result of both low-pass and high-pass synthesis filters are added to reconstruct the original signal. This process can be completely lossless using the (5,3) integer filter taps provided by LeGall or lossy but more effective with Daubechies (9,7) floating point coefficients. The DWT can easily be expanded to two dimen-
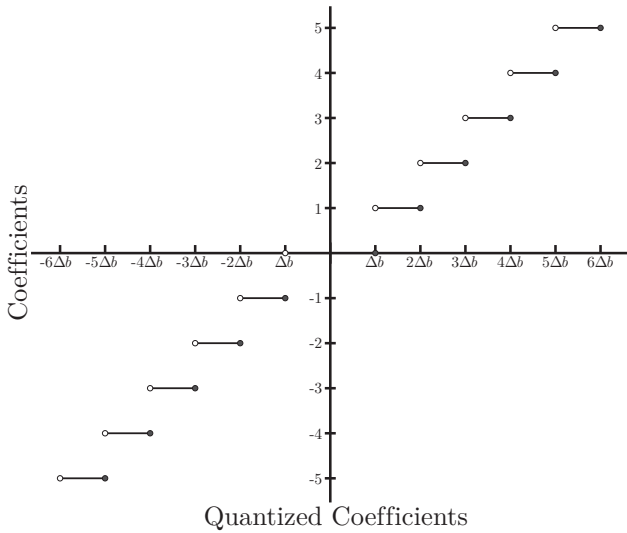

Figure 3: Uniform quantizer with central dead zone

sions by successively applying analysis filters on horizontal and vertical axis producing four subbands: low-pass on both axis (LL), horizontal high-pass and vertical low-pass (HL), horizontal low-pass and vertical high-pass (LH), and high-pass on both axis (HH). After this decomposition, LL corresponds to a smaller version of the original image that can be decomposed further by reapplying the process. For instance, if three levels of decompositions are required (see

Fig. 2) the first subbands are labeled HL1, LH1 and HH1. LL1 is further decomposed producing HL2, LH2 and HH2. This process is repeated one more time on LL2. LL3 is often referred as LL because in the end only one LL subband is left. Just as discrete Fourier transform can be heavily optimized (**?**) with fast Fourier transform algorithms, DWT computations are not performed by traditional convolution but with a "lifting scheme" that significantly reduces computational complexity and provide in place computation thus reducing memory requirements while producing the exact same results.

2.1.3 Quantization

JPEG2000 quantization is simple as it uses a uniform quantizer with a central dead zone. This means that approximation steps are equally spaced (Δb) except around zero where it is twice as large (see Fig 3). When lossless compression is required, the DWT is performed on an integer-to-integer basis and the step size is set to one ($\Delta b = 1$), otherwise it can be configured independently for each subband of each transformation level explicitly or inferred from the size specified for the LL subband. Usually the step size is kept very small to allow efficient rate distortion optimization at the codestream organization phase.
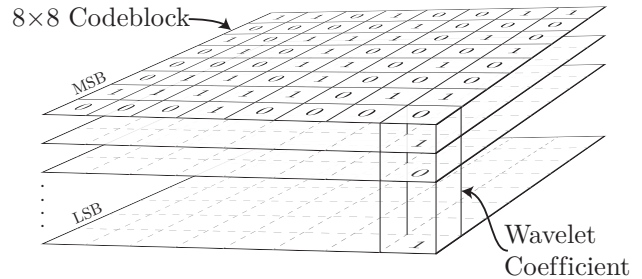
2.1.4 Entropy coding(Tier-1 coding)

Figure 4: Bit-plane coding

Entropy coding in JPEG2000 performed by a bit-plane binary arithmetic coder called MQ-Coder. Using this algorithm, coefficients of the DWT are divided in rectangular areas, called code-blocks, with power of two (2^n) dimensions (32x32 is common). Code-block dimensions are uniform across all subbands and resolution levels. They are entropy coded independently to allow random spatial access as well as better error resilience. Each code-block is further decomposed into bit-planes that are sequentially encoded (Fig. 4) from the most significant bits to the least significant bits. This ensures that the most relevant information across all coefficients of the code-block is encoded first. Bit-planes are encoded in three passes (significance propagation, refinement and cleanup). Each coding pass serves as valid truncation point for rate distortion optimization. Decoding only few passes produce a coarse approximation of the original coefficients and, as a result, of

the original image; adding more passes further refines the outcome thus reducing distortion.

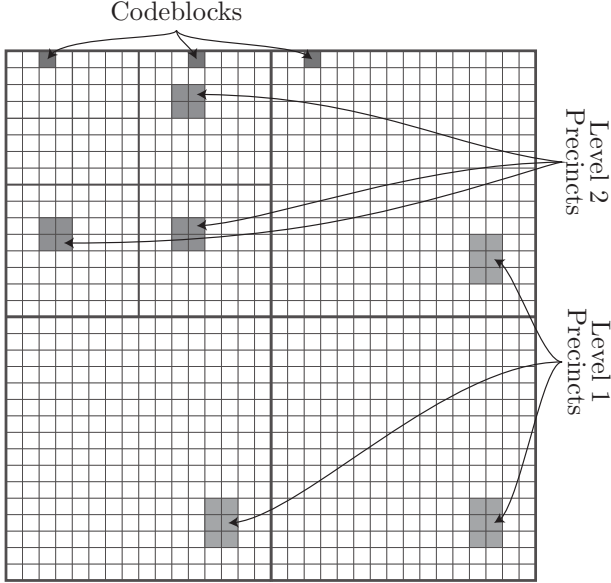### 2.1.5 Code-stream organization (Tier-2 coding)

Figure 5: code-blocks and precinct organization

Coefficients are further organized (Fig. 5) in precincts that include neighbouring code-blocks from every subbands of a given resolution level needed to decode a spatial region from the original image. Their dimensions are also power of two (2^n) and they must be equal or larger than the code-blocks. They represent a space-frequency construct that serves as a building block for random spatial decoding. Bit-planes coding passes are distributed across layers corresponding to quality increments over the entire image. Each layer includes contributions from every code-block from all components and subbands. Bit-planes passes included in a given layer are not necessarily the same for all code-blocks since they are selected as part of the post compression rate-distortion optimization (PCRD-opt).

Packets are the last organizational elements. They are the fundamental code-stream building blocks and contains bit-planes coding passes corresponding to a single quality layer of a given precinct. They can be arbitrarily accessed and they are the construct that enables advanced features of JPEG2000 such as resolution scalability, progressive quality decoding and random spatial access. Packets can be ordered in the code-stream to allow progressive decoding along four axes: resolutions, quality layers, component and position. When progression along the quality axis is required, packets representing the most significant bits for all components across all resolutions and precincts would be placed at the beginning of the file. When the image is downloaded, the most significant bits from every coefficient will arrive first. They can be decoded to produce a low quality preview that is progressively refined as more packets are received. On the other hand, if resolution progression is needed for a three decomposition level image, packets from all layers, components and precincts from LL3, HL3, LH3 and HH3 subbands are placed at the beginning of the file, followed by HL2, LH2 and HH2, and finally HL1, LH1 and HH1. This technique ensure that packets are already in the desired decoding progression order when images are transmitted over low bandwidth networks enabling customizable refinements as more data is transmitted.

Rate control can be achieved in two ways in JPEG2000: quantization steps can be specified for each subband of each resolution level or the quantization steps can be kept very small for post compression rate distortion optimization (PCRD-opt). The first technique is quite similar to what was found in JPEG. PCRD-opt, on the other hand, involve computing both distortion and rate (bytes needed) for each possible truncation point of every code-blocks. Because distortion measurements, usually mean squared error (MSE), are additive, Lagrangian optimization can be performed to achieve optimal rate for a specified distortion or optimal distortion for a given rate.
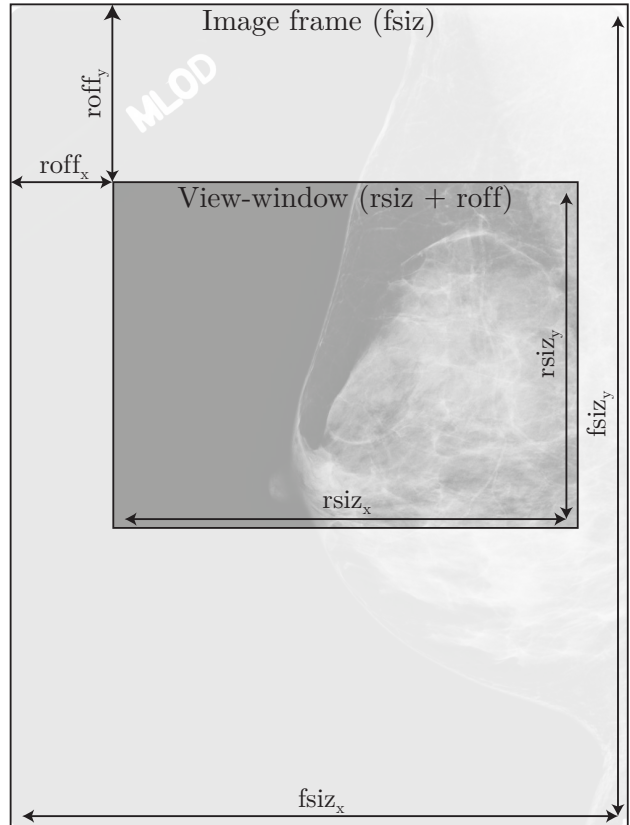
2.2 Image streaming with JPIP

Figure 6: JPIP View-Window

Traditional image transfer methods, such as plain HTTP, can't fully exploit JPEG2000's flexible code-stream organization. Because files are downloaded sequentially, pro-

gressive decoding and rendering can only be performed in the order that was set at the encoder when packets were arranged. The JPEG2000 Interactive Protocol (JPIP) was developed to solve this issue by defining a standard communication protocol that enables dynamic interactions. Streaming can be based on tiles (JPT-stream) or precincts (JPP-stream) when finer spatial control is required. In JPP-stream mode, images are transferred in data-bins that contain all packets of a precinct for the required quality layers. Requests are performed using a view-window system (Fig. 6) defined by frame size (fsiz), region size(rsiz) and offset(roff). These parameters can be used to retrieve image sections at suitable resolutions. The request can also include specific components (comps) and quality layers (layers). As an example, if the screen resolution is 1024×768 and image size is unknown, the client could issue a JPIP request with

```
fsiz=1024,768&rsiz=1024,768&roff=0,0&layer=1
```

to retrieve the first quality layer of the image at a resolution that best fits the display area. On the other hand, if the upper right corner of the image is required with 3 quality layers, the request would be:

```
fsiz=2048,1496&rsiz=1024,768&roff=1024,0&layer=3
```

Because clients have no a priori information (number of layers, image size, tile or precinct size, etc.) about the requested images, servers can slightly adapt incoming request. For instance, server implementations can redefine requested regions so their borders correspond to those of precincts or tiles. In the end, a JPIP enabled HTTP server can easily and effectively enable the same flexibility and interactivity that is available from a locally stored JPEG2000 file.

2.3 Image storage and communication with DICOM

Digital Imaging and Communications in Medicine (DICOM) is the leading standard in medical imaging. Work started almost thirty years ago (?), in 1983, as joint effort between National Electrical Manufacturers Association (NEMA) and the American College of Radiology (ACR) to provide interoperability across vendors when handling, printing, storing and transmitting medical images. The first version was published in 1985 and the first revision, version 2.0, quickly followed in 1988. Both versions only allowed raw pixel storage and transfer. In 1989, DICOM's working group 4 (WG4) that was tasked with overseeing adoption of image compression, published its recommendations in a document titled "Data compression standard" (?). They concluded that compression does add value and defined a custom compression model with multiple optional prediction models and entropy coding techniques. Unfortunately, fragmentation caused by many implementation possibilities meant that while images were compressed internally when stored, transmission over networks was still performed with uncompressed raw pixels to preserve interoperability.

DICOM 3.0 was released in 1993 and it included new compression schemes: the JPEG standard that was published the year before, Run Length Encoding and the pack bits algorithm found in the Tagged Image File Format (TIFF). In this revision, compression capabilities could also be negotiated before each transmission allowing fully interoperable lossy and lossless compression.

In the mid-90s, awareness increased around new compression techniques based on wavelets. At the time, they offered flexible compression scalability and higher quality at low bit rates but no open standard format was available causing interoperability issues.

2.3.1 DICOM with JPEG2000

The base JPEG2000 standard was finalized at the end of 2000 and DICOM supplement 61: JPEG2000 transfer syntax (?) was adopted in 2002. The standard did not address compression parameters or clinical issues related to lossy compression, but defined two new transfer syntaxes; one for lossy&lossless and one for lossless. Multi-component transformation (MCT), part of JPEG2000 extensions (part 2), was adopted in supplement 105 (?) in 2005.

3D volumes or image stacks can be decorrelated in the same way RGB images are with component transforms. JPEG2000 allows up to 16,384 components. Typical colour images only use three, but with volumetric data, such as CT scans, each slice can be represented as a component. Two types of decorrelation techniques can then be applied: an array-based linear combination (e.g. differential pulse-code modulation (DPCM)) or a wavelet transform using the same analysis filter on the Z axis that is already used by encoder on the X and Y axes. Using the later technique lossless compression efficiency can be improved by 5-25% (?). Both techniques reduce random spatial access capabilities of the codec since multiple components are required to reverse this inter-component transform. This effect can be mitigated with component collections (slice groups) independently encoded and stored as separate DICOM fragments, but at the cost of reduced coding efficiency.

2.3.2 DICOM with JPIP

Acknowledging the advantages of web services on productivity and quality of care, DICOM adopted in 2004 supplement 85 titled "Web Access to DICOM Persistent Objects (WADO)" that enabled easy retrieval of DICOM objects trough HTTP using URL. Similarly, JPIP was adopted latter in 2006 as part of supplement 106 (?) to enable access to DICOM images without having to wait for the reception of all pixel data. Applications of JPIP include navigation of large image stacks, visualization of a single large image and use of thumbnails. Implementation and interoperability are easy to achieve because of the transfer syntax negotiation process introduce in DICOM 3.0. When both

devices are JPIP ready, pixel data from DICOM files are simply replaced by HTTP/S JPIP URL and the transfer syntax is changed accordingly.

Unfortunately, JPIP doesnt know about multi-component transform of 3D volumes. In that case, clients must decide, on their own, which data is required. This issue was addressed with JPEG2000 part 10 (JP3D) which has not yet been included in DICOM.

3 PREVIOUS WORK

JPIP has applications in fields other than diagnostic imaging. With remote sensing, for instance, satellites can capture very high resolution images. In (?), JPIP is used to stream images of the sun taken from NASA's solar dynamic observatory. These images are up to 16 megapixels in size and contain 8 spectral channels. In their experiments, users could quickly look through a day's worth of data at moderate spatial and temporal resolutions (1000×1000 at 5min intervals, 0.5bits/pixel) using only 66MB while the original dataset was 600GB (4096×4096 at 10 sec intervals). Using this technique, full resolution images can be streamed when regions of interest are identified. Their implementation also allowed for playback at 25 frames per second and they expect better performance after optimization. These results suggest that navigation of large CT image stacks could benefit from JPIP streaming.

JPIP is also being used in microscopy (?) to stream whole slide images. These images are truly immense; slides can be as large as 30×20 millimeters and can be captured at resolutions as small as $0{,}2\ \mu\text{m}/$pixel, thus producing $150{,}000px \times 100{,}000px$ images. These 15 gigapixels images are so big that they even exceed theoretical limits of DICOM (16bit row/column addressing or 65,565). Uncompressed, they can weigh up to 40GB (DICOM files size limit is 4GB) each, but it is very unlikely all regions at the highest resolution are required for analysis. Image compression and interactive streaming are therefore essential. The authors were able to implement a DICOM-compatible JPIP solution for remote examination and they expect JPEG2000 to become a widely-accepted standard in their field. Their experiments showed that images of virtually any size can be efficiently streamed.

In radiology, JPIP was also used to allow thin devices (PDA, tablets, etc) to access properly scaled images trough an hybrid DICOM-JPIP solution (??). To preserve compatibility with existing devices, clients performed JPIP queries as private data tags within the DICOM communication protocol. Imaging archives unaware of JPIP would simply ignore these queries and return the complete unaltered images. Compatible servers, on the other hand, would process the query and provide a smaller image well suited for display on the thin device. They argue that providing JPIP trough DICOM instead of HTTP would make interoperability easier. However, image streaming wasn't implemented or tested and we will show how JPIP can be used in a completely standard compliant fashion without relying on undocumented private data tags.

Security concerns were raised in (?) because JPIP could expose pixel data as a simple HTTP URL. However, this issue was addressed in (?) where they proposed the use of features already provided by JPEG2000 and DICOM. With their implementation, DICOM files containing JPIP URLs are encrypted as per DICOM part 15, Security Profiles, and the JPEG2000 stream is ciphered. Decoding the image cipher now requires a key that is sent in the encrypted DICOM document as a private tag. Part 10 of the JPEG2000 standard, JPSEC, addresses security and codestream ciphering.

Finally, a pilot project using XDS-I enabled electronic health record system with embedded JPIP was recently implemented in Shanghai (?). Four hospitals were connected through a grid peer-to-peer network with interconnection bandwidth as low as 250KB/s. At this rate, users could still display computed radiography previews within less than one second. Computed radiography display at screen resolution took less than 3 seconds while the entire image could be downloaded in 8 seconds. Their goal was to enable remote diagnostic from a large regional hospital for patients admitted at three other smaller community hospitals that do not have full-time radiologists. They have received positive feedback from the users at the small community hospital as well as from radiologists of the regional hospital. Their solution is based on XDS-I and is similar to what is presented in the next section. However, they have implemented only one visualization scheme: a low-resolution image is downloaded first and subsequent resolutions are progressively downloaded. Spatial random access is therefore impossible and different quality layers are not available. In the later sections, we will present other visualization schemes that are more effective.

4 INCLUDING IMAGES IN THE EHR

4.1 XDS Architecture

Most care giving organizations such as hospitals, radiology clinics or ambulatory care are supported by multiple systems that may produce, store or retrieve clinical artifacts. The XDS architecture allows these artifacts from different care delivery systems to be shared in the form of documents that represent units of health information being shared in a standard format. XDS is therefore a document centric architecture that can be used to publish, query and retrieve specific clinical documents for a specific patient.

XDS's architectural model is based around a central registry that holds metadata describing every published document and can respond to queries about artifacts meeting specific criteria. Registries don't store documents themselves, but maintain information about locations where they can be retrieved called document repositories. A registry would typically store document metadata belonging to multiple repositories. Systems that produce artifacts relevant to patient's continuity of care, document sources,
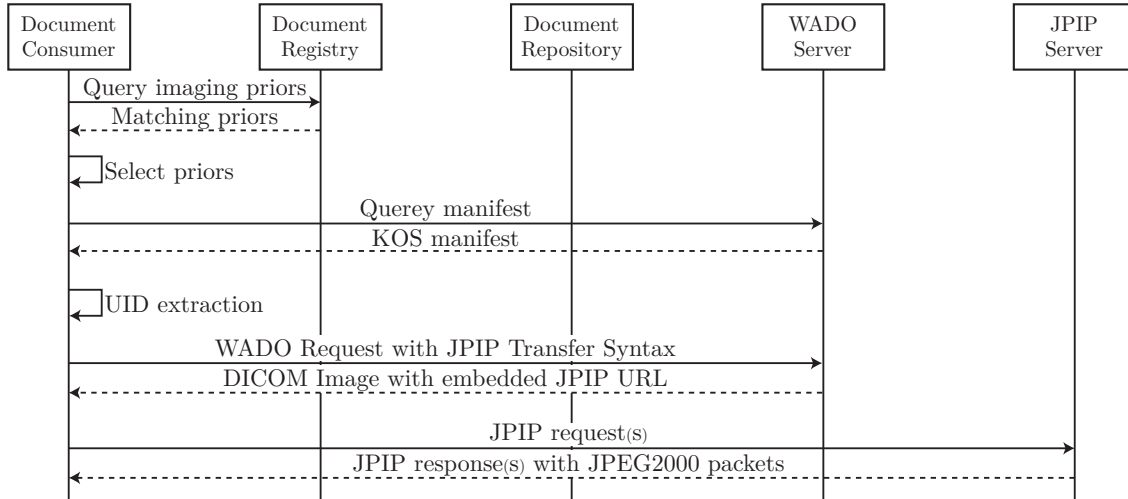
Figure 8: XDS-I with JPIP sequence diagram

are responsible for publishing documents. Systems that are interested in accessing patient's records, document consumers, will simply query the registry using metadata. Registries will include each document address within query responses, allowing consumers to retrieve documents from their repositories.

Digital images are increasingly important components of a patient medical record and XDS-I, an extension to XDS shown in Fig 7, was especially designed to enable sharing of diagnostic imaging. To publish a set of images, a DICOM manifest containing references to one or several DICOM instances is created by the sources and subsequently stored in the document repository and registered at the document registry. With this solution, only the manifests are published to the repository; not the images themselves. When a consumer retrieves a manifest it is decoded to obtain the list of referenced images. Each referenced object in those manifests are specified using Universal Identifiers (UID) along with an application title (AEtitle) that is used to identify which document source stores the actual image files. The consumer can then retrieve the required image using traditional DICOM transaction: C-MOVE or Web Access to DICOM Persistent Objects (WADO).

4.2 Delivering images using JPIP

JPIP is a client/server standard image streaming protocol that allows client applications to request only portions of a JPEG 2000 image that are necessary to fulfill the client's viewing needs resulting in improved bandwidth efficiency. This improvement is significant for medical applications as medical images are often either very large images or bundled in very large sets. JPIP is also valuable for diagnostic imaging because it relieves the client application from importing the image into its environment thus eliminating the problems of persistency, consistency and reconciliation. JPIP can easily be integrated in a XDS-I environment using these steps while retrieving artifacts (Fig. 8) (**?**) :

1. An imaging workstation (XDS-I consumer) queries the registry using metadata for imaging priors that are relevant for the case at hand. The registry responds with a list of manifests, each representing a set of images that are available from an imaging archive (XDS-I source).

2. The XDS-I consumer selects and retrieves a specific manifest from the document repository.

3. For each referenced DICOM instance within the manifest, UIDs are extracted and used to form a DICOM
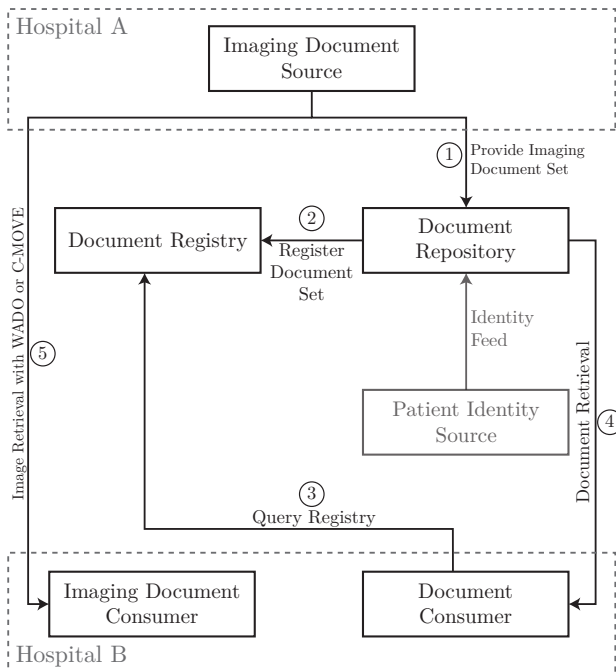

Figure 7: XDS-I Architecture

WADO query that will be sent to the image source. WADO queries are web requests that are used to download specific DICOM instance over HTTP:

```
http://server.com/WadoService?requestType=WADO
&studyUID=1.2.250.1.59.40211.1234
&seriesUID=1.2.250.1.59.40211.789
&objectUID=1.2.250.1.59.40211.262.2
&contentType=application%2Fdicom
&transferSyntax=1.2.840.10008.1.2.4.94
```

The first three parameters are DICOM UIDs for study, series and instance. The two following parameters set the return type to a DICOM file and ask for specific transfer syntax. This is typical of any XDS-I retrieve operation. The difference is that JPIP aware consumers can use the transfer syntax to explicitly request DICOM-JPIP files. If this transfer syntax is not supported by the document source, uncompressed images are returned thus maintaining compatibility.

4. The DICOM-JPIP document returned to the consumer by JPIP enabled source will include a pixel data provider URL that specifies the address of the JPIP server capable of providing the actual image data. The Consumer application is now free to perform JPIP requests allowing interactive visualization. Typical JPIP requests are as follow:

```
http://server.com/JpipService?target=image.j2k
&fsiz=800,600&rsiz=800,600&roff=0,0&layers=3
```
The server should reply with an image properly scaled for an 800×600 display with the first three layers of quality that were defined at the encoder.

Enhancing the user's experience

JPIP has already proven capable of handling very large images with whole slide microscopy. Security and compatibility issues have also been addressed to some extent in previous work and a JPIP enabled XDS-I pilot has been successively implemented in Shanghai.

As we all know, image compression can significantly improves network and storage efficiency. JPIP can enable further gains by downloading only diagnostically important sections at full quality. However, the biggest advantages of JPIP based image streaming are the improvements to the user's experience and productivity. Instead of waiting several minutes for the images to be imported in the local repository, users can start remote visualization within seconds. Physicians can become much more productive because the sections they want to investigate are displayed much faster while regions of little diagnostic value are displayed with low resolution, low quality or are simply not downloaded.

To illustrate some possibilities offered by JPIP, we have considered the case of a large, 4728×5928, 57 megabytes digital X-ray mammography displayed on a typical 1080×1920 pixels consumer level monitor using two different visualization schemes: "zoom & pan" and a magnifying lens.

The first method corresponds to the typical magnifying glass behavior where the full screen displays a magnified region centered on the selected pixel and the magnified region can be move with a "hand" tool. With the later scheme only, a small 512×512 window shows the full resolution surrounding the selected pixel. The user is able to drag this window with the mouse, but the entire image is never zoomed. In both cases, a preview is downloaded first to best fit the screen and subsequent packets are queried according to the visualization scheme. To compare both schemes, we will consider a single region of interest represented by C in Figure 9(a).

A modified version of the OpenJpeg library was used to compress the image, capture packets information and simulate JPIP transactions. JPIP query string length and HTTP overhead are not taken into account, but provided that session and cache model defined in the JPIP standard are enabled they should represent less than 200 bytes per transactions. All packet transfer time are calculated based on 2 megabits per second effective transfer rate and latencies are ignored. At that speed, the original image would take 224 seconds to download while the losslessly compressed JPEG2000 DICOM file would take 34 seconds. Matlab is used to calculate Peak Signal-to-Noise Ratio (PSNR) for each region using these expressions:

$$RMSE\left(I, \hat{I}\right) = \sqrt{\frac{1}{s_x s_y} \sum_x \sum_y \left(\left(I_{(x,y)} - \hat{I}_{(x,y)}\right)^2\right)}$$

$$PSNR\left(I, \hat{I}\right) = 20\log_{10}\left(\frac{I_{\max} - I_{\min}}{RMSE\left(I, \hat{I}\right)}\right)$$

Efficiency of each visualization technique is highly dependent on packet organization of the JPEG2000 codestream. Precinct sizes, quality layers and decomposition levels are all parameters that affect the degree of interactivity that can be achieved with JPIP. These settings have to be carefully adjusted to meet visualization requirements of specific modalities. Computed tomography, for instance, produce large sets of small images; spatial random access is therefore useless, but fast previews are needed to enable stack navigation and thumbnails are frequently used. In that case, precincts are not needed and should not be used; quality layers should be configured to enable fast preview and enough decomposition levels are required to enable thumbnails. Digital mammography images have completely different requirements; they are very large and require good spatial visualization. In our experiments, we chose a five level decomposition to enable good resolution scalability with the smallest resolution, LL, being 371×296. code-block size was set to 64×64 and the 5/3 lossless integer-to-integer wavelet transform was used. Precincts were 256×256 for level 1, 128×128 for level 2 and 64×64 for level 3, 4 and 5 to support flexible random spatial access. This means that precincts from resolution 1, 2 and 3 all represent 256×256 area of the full resolution image. In total, 456
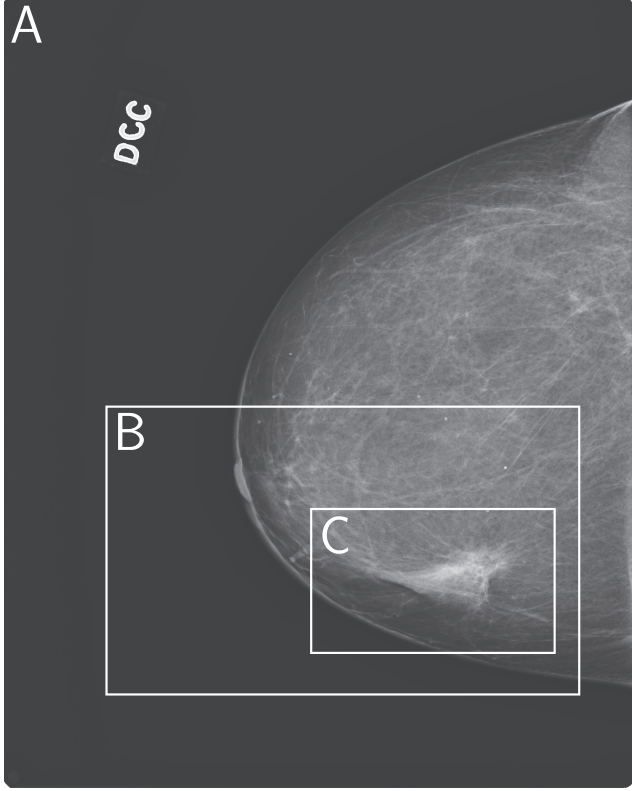
(a) Zoom & pan visualization scheme
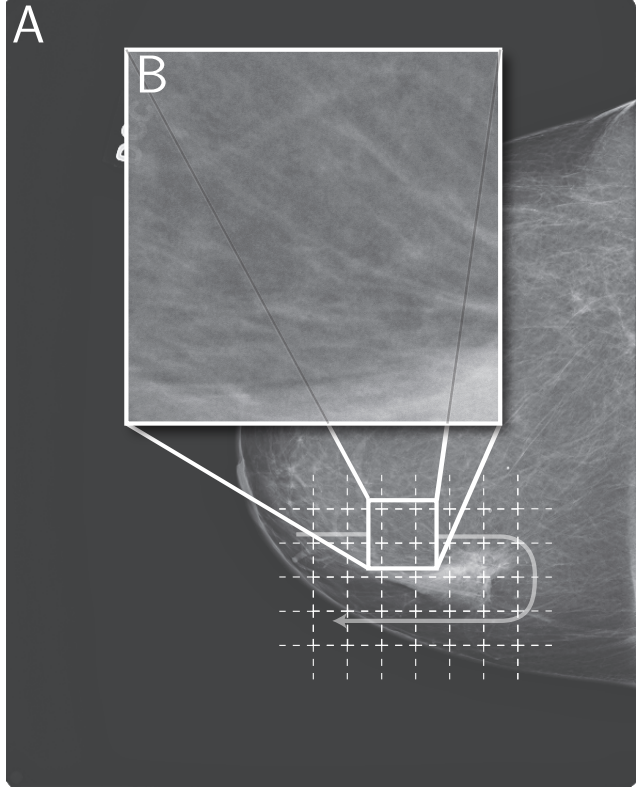
(b) Magnifying lens visualization scheme

Figure 9: Visualization schemes

($\lceil 5928/256 \rceil \times \lceil 4728/256 \rceil = 24 \times 19$) precincts are created for these resolutions. Figure 9(b) shows the size and position of some precincts with white dashed lines. Four quality layers where used : 100:1 to provide very fast previews, 25:1 which is the recommended maximum CR for breast digital radiography by the Canadian Association of Radiologists (CAR)**?**, 10:1 and lossless. Images are losslessly compressed since physician should be able to access unaltered images, if time and bandwidth allows, and also because the FDA doesn't allow lossy compression of digital mammogram**?**. Once compressed, the JPEG2000 file contains 8,527,709 bytes. The compression ratio is about 1:6.5, but packets are organized to enable streaming based on the other configured CRs. The post compression rate distortion optimization allocates bits across all precincts to minimize MSE for a given rate considering all resolution decomposition. This means that, in our case, requesting only the first quality layer for a resolution other than the full resolution or for a region other than the full image may not result in 100 time reduction compared to downloading the same resolution of region losslessly. Furthermore, bits are unevenly spatially distributed and precincts representing the almost completely black background need very few bytes compared to precincts of the breasts. Figure 10 shows precincts with under 5 kilobytes of combined data for resolution 1, 2 and 3 under the highlighted region. The remaining precincts represent 96.6% of the data but only covers 45% of the area.

With the "zoom & pan" visualization scheme, a low resolution version of the image (A) is first downloaded and quickly displayed. The user is then allowed to successively enlarge the image, one screen at a time (B), until the original resolution (C) is displayed as shown in Fig. 9(a). For the first step, the resolution reduction that best fits the image (4728×5928) in the viewing area (1920×1080) is three ($[4728, 5928]/2^3 = [591, 741]$). Interpolation is then used to best fit the screen area. To provide a good user experience and keep the physicians productive while the image is being downloaded, the first quality layer, equivalent to a 100:1 compression ratio, is transfered and displayed as quickly as possible. This represents a $82,994$ bytes transfer that should take 0.3 seconds at 2 megabits per second. Even considering overhead, latencies and decoding this should take well under a second and the physicians will be able start their investigation by looking for regions of diagnostic significance while the following quality layers are successively downloaded and decoded. The process continues until the image is displayed losslessly or until the user issues a zoom command. In our example, 163,805 bytes or 0.65 seconds are required to download a lossless version of the preview image.

When the user zooms around a region of interest, the image is enlarged to the closest resolution reduction that enables filling the entire screen without using any interpolation. In our case, a (3840×2160) is requested with one decomposition reduction. Resolution 5 and 4 are prob-

PSNR is computed on the displayed region and considers the worst case where previous resolutions were not losslessly downloaded.

Table 1: Transfers for magnifying zoom & pan scheme

Region	Res	QL	PSNR	Step Size	Total Size
A	3	1	54.43	82,994	82,994
	3	2	57.72	13,834	96,828
	3	3	64.78	35,030	131,858
	3	4	Inf	31,947	163,805
B	1	1	39.34	104,382	268,187
	1	2	42.35	81,697	349,884
	1	3	51.62	242,094	591,978
	1	4	Inf	455,733	1,047,711
C	0	1	33.67	3,929	1,051,640
	0	2	37.33	139,394	1,191,034
	0	3	46.43	293,544	1,484,578
	0	4	Inf	762,149	2,246,727

With the magnifying lens visualization scheme, a small preview is downloaded first as in the previous case, but the physician then use a draggable window to view a small 512×512 region at full resolution. Figure 9(b) shows the window (B) that would normally be centered on the expanded region, but was translated for clarity.

Table 2: Average transfers for magnifying single precinct

QL	Bytes(Avg.)	Time(sec)	Bytes(Max)	Time(sec)
1	1,273	0.005	4,301	0.017
2	4,979	0.020	11,788	0.047
3	9,922	0.040	19,326	0.077
4	23,428	0.094	36,148	0.145
ALL	39,604	0.158	55,700	0.223

The preview (A) is first progressively downloaded up to lossless in 0.65 seconds as like before. When the physisian uses the magnifying glass for the first time, nine precincts are required. Subsequently, when the window is moved, only three new precincts are usually revealed. Table 2 shows the average, excluding the highlighted region of figure 10, and worst download sizes along with the time needed to download a single precinct. Downloading nine precincts losslessly would take 1.5 seconds on average, but intermediate previews can still be displayed very quickly with a 25:1 version in 0.25 second on average.

To compare with the "zoom & pan" visualization scheme table 3 shows transfers for 12 groups of 4 precincts covering the same area. Precincts were only grouped two-by-two, 512×512 pixels, for PSNR computation and presentation. Using the magnifying lens visualization techniques, the same exam can be performed with only 1,738,150 bytes and the total download time is now under 7 seconds. This 25% gain is due to the absence of the intermediate step shown by region B of Fig. 9(a). This visualization scheme may seem slightly unnatural, but it provides faster interactions and consumes less bandwidth because relatively small regions are query following each user action.

Figure 11 shows a comparison of total transfer time for the four considered schemes. Diagnosis can begin in under
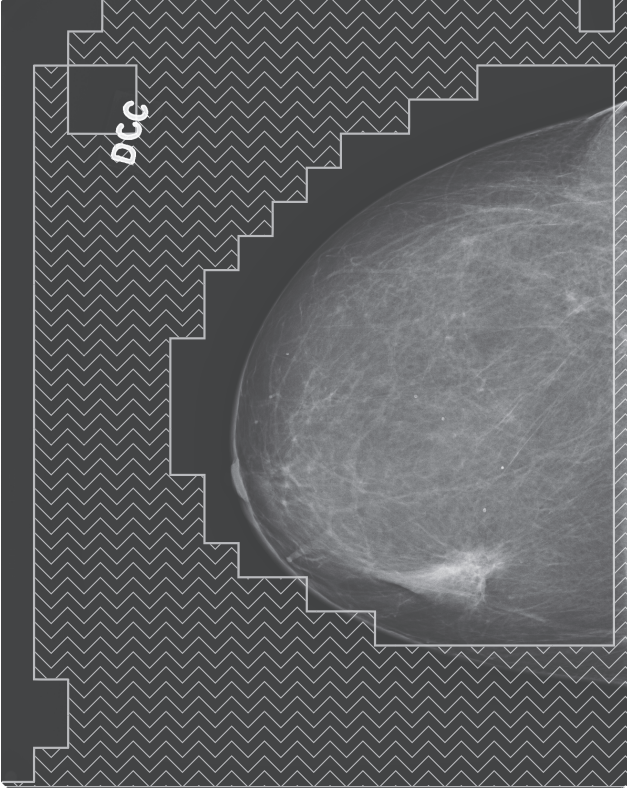

Figure 10: The highlighted region represent 55% of all level 1,2 and 3 precincts but only 3% of the file size

ably already completely downloaded as part of the preview, but precincts from resolution 3 and 2 are required. 135 256×256 precincts ($\lceil 3840/256 \rceil \times \lceil 2160/256 \rceil$) need to be downloaded from each resolution. As before, only one quality layer is queried at first and 104,382 bytes are downloaded in 0.4 seconds. At this point, only 30%(135/456) level 2 and 3 precincts contribute to the image. The viewing area is refreshed three more times within 4 seconds until the region is displayed losslessly. No interpolation is required because the image is already sub-sampled by a factor of 2.

The process is repeated when the user zooms again to retrieve the full resolution. An image of sufficient quality for diagnostic (25:1) is presented to the physician in less than 2 seconds and is subsequently displayed losslessly within 3 seconds. Only 9% (40/456) of level 1 precincts are required to decompress this image region. In total, 2,246,727 bytes are required for this use case, but the physician never waits for more than 3 seconds before the image is refined. The total download time is about 9 seconds. Furthermore, only 426,230 bytes (about 2 seconds of total download time) would be required in total to perform this diagnostic using the compression recommendations from the Canadian Association of Radiologists for breast digital radiography (25:1). Table 1 shows the details for each step for regions A,B and C. Resolution (res) is presented in the form resolution reductions ($size' = size/2^{Res}$) and quality layers (QL) are 1=100:1, 2=25:1, 3=10:1 and 4=lossless. The
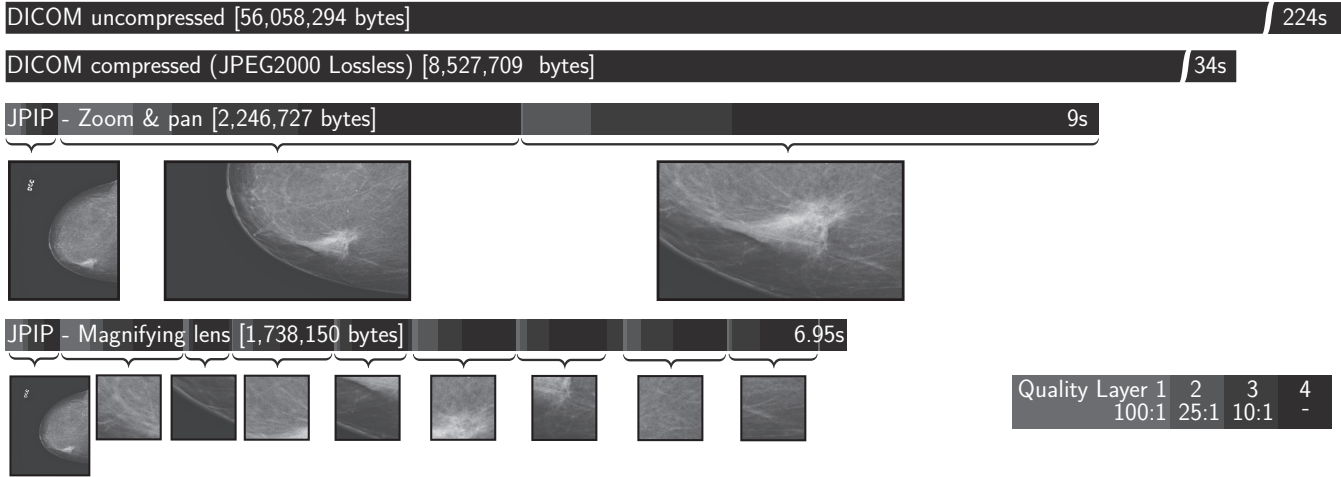
Figure 11: Download size and time for each visualization scheme

a second with both JPIP schemes while importing the image in the local repository requires complete transfer before the exam can begin. Furthermore, Software clients could implement prediction models to anticipate which regions the physician is likely to review next and prefetch those in the background if bandwidth enables even faster interactions. Monitor and image resolution are continuously increasing but the discussion here will always be valid as long as screen resolution is smaller than image dimensions.

CONCLUSION

JPIP brings two major advantages when viewing medical images in distributed environments, such as the ones encountered with EHR. The first is the streaming capability that eliminates the need for importing foreign images into a medical image archive, thus avoiding the problems related to information consistency and persistency management of duplicated images. The other is the significant improvement in bandwidth efficiency and user experience when viewing medical images that are usually either large or part of very large image sets.

JPIP enables client applications to visualize large images at the best screen resolution with much less data than required when visualizing the same image losslessly compressed. It also enables visualization of low quality images very quickly while enabling progressive refinements at subsequent moments. In this paper, we proposed an approach to compress images with JPEG2000 and implement JPIP in order to visualize large digital mammography with two common schemes: "zoom & pan" and magnifying lens. Enabling JPIP with any old JPEG2000 file is not enough, streaming performances and visualization possibilities are highly dependent on compression parameters. Decomposition levels, precinct sizes and quality layers all play an important role in enabling visualization scenarios and improving user experience. Requirements introduced by both visualization schemes were discussed and compression parameters that are well suited for their application were proposed. We have also measured bandwidth, productivity and user experience improvements that are enabled by both visualization schemes. We showed a detailed evaluation of bandwidth requirements of both schemes and compared them with losslessly compressed and uncompressed image imports. Our experiment showed that JPIP does significantly improve user experience by allowing physicians to start their investigations within seconds and stay productive during the entire download process.

While using JPIP to deliver medical images from the EHR to the radiologist's workstation appears very promising, many challenges still exist: image display applications need to integrate a JPIP client; image archive systems need to integrate a JPIP server; adoption is needed and interoperability testing is required. Fortunately, DICOM's transfer syntax negotiation allows client and server vendors to implement JPIP at their own pace since incompatible implementations will simply fall back to traditional DICOM files. IHE's cross enterprise sharing for imaging integration profile already covers DICOM files with embedded JPIP URL, but to our knowledge, no test cases are currently available. Also, JPIP is useless if compression parameters are not carefully configured with a specific visualization scheme in mind. Future work should include developing new JPIP test cases and trans-coding techniques that would enable efficient streaming of any image and help foster wider adoption.

Table 3: Transfers for the magnifying lens scheme

Region	Res	QL	PSNR	Step Size	Total Size	Time
-	3	4	Inf	163,805	163,805	0.66
1	0	1	27.72	6,538	170,343	0.68
	0	2	31.27	28,774	199,117	0.80
	0	3	41.83	63,530	262,647	1.05
	0	4	Inf	112,714	375,361	1.50
2	0	1	34.86	258	375,619	1.50
	0	2	35.72	392	376,011	1.50
	0	3	39.62	11,019	387,030	1.55
	0	4	Inf	71,606	458,636	1.83
3	0	1	7.87	12	458,648	1.83
	0	2	14.1	12	458,660	1.83
	0	3	25.67	12	458,672	1.83
	0	4	Inf	727	459,399	1.84
4	0	1	28.88	10,540	469,939	1.88
	0	2	33.85	41,144	511,083	2.04
	0	3	43.86	56,620	567,703	2.27
	0	4	Inf	111,749	679,452	2.72
5	0	1	35.59	1,333	680,785	2.72
	0	2	38.59	8,657	689,442	2.76
	0	3	44.7	27,049	716,491	2.87
	0	4	Inf	115,999	832,490	3.33
6	0	1	41.13	26	832,516	3.33
	0	2	43.55	22	832,538	3.33
	0	3	46.86	121	832,659	3.33
	0	4	Inf	6,332	838,991	3.36
7	0	1	29.64	13,073	852,064	3.41
	0	2	34.69	40,705	892,769	3.57
	0	3	44.61	55,916	948,685	3.79
	0	4	Inf	111,747	1,060,432	4.24
8	0	1	32.65	3,110	1,063,542	4.25
	0	2	35.3	14,042	1,077,584	4.31
	0	3	43.82	48,822	1,126,406	4.51
	0	4	Inf	118,928	1,245,334	4.98
9	0	1	36.76	69	1,245,403	4.98
	0	2	38.01	76	1,245,479	4.98
	0	3	40.65	717	1,246,196	4.98
	0	4	Inf	32,772	1,278,968	5.12
10	0	1	25.11	5,328	1,284,296	5.14
	0	2	28.31	28,506	1,312,802	5.25
	0	3	39.59	68,883	1,381,685	5.53
	0	4	Inf	112,050	1,493,735	5.97
11	0	1	27.56	1,293	1,495,028	5.98
	0	2	29.1	9,024	1,504,052	6.02
	0	3	38.4	60,164	1,564,216	6.26
	0	4	Inf	117,205	1,681,421	6.73
12	0	1	33.76	49	1,681,470	6.73
	0	2	34.73	68	1,681,538	6.73
	0	3	37.11	1,046	1,682,584	6.73
	0	4	Inf	55,566	1,738,150	6.95

REFERENCES

American College of Radiology (2007). ACR technical standard for electronic practice of medical imaging. Technical report.

Canada Health Infoway. Diagnostic Imaging Benefits Evaluation Final Report. Retrieved April 24, 2011, from Canada Health Infoway web site: `https://www2.infoway-inforoute.ca/Documents/CHI%20Diagnostic%20Imaging%20Report_ENGintro.pdf`.

CAR (2008). CAR Standards for Irreversible Compression in Digital Diagnostic Imaging within Radiology. Retrieved December 4, 2010, from the Canadian Association of Radiologists web site: `http://www.car.ca/uploads/standards%20guidelines/Standard_Lossy_Compression_EN.pdf`.

Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. *Communications on pure and applied mathematics*, 41(7):909–996.

Dinu, D. and Ivetic, D. (2009). Architectures of DICOM based PACS for JPEG2000 medical image streaming. *Computer Science and Information Systems*, 6(1):185–203.

Dragan, D. and Ivetic, D. (2008). An Approach to DICOM Extension for Medical Image Streaming. In *Proceedings of the 19th DAAAM International Symposium," Intelligent Manufacturing & Automation", Trnava, Slovakia*.

Haggag, A., Abdelwahab, A., Ibrahim, A., Lu, J., and Yahagi, T. (2007). Interactive Secure Dissemination of Medical Images Using JPEG 2000. In *Information Technology Applications in Biomedicine, 2007. ITAB 2007. 6th International Special Topic Conference on*, volume 00, pages 103–106. IEEE.

Ivetic, D. and Dragan, D. (2011). Medical image on the go! *Journal of Medical Systems*, 35:499–516.

Lee, D. (2005). JPEG 2000: Retrospective and New Developments. *Proceedings of the IEEE*, 93(1):32–41.

Macedo, D. D. D., Wangenheim, A. V., Dantas, M., and Perantunes, H. G. (2009). An architecture for DICOM medical images storage and retrieval adopting distributed file systems. *International Journal of High Performance Systems Architecture*, 2(2):99–106.

Mallat, S. (1989). A theory for multiresolution signal decomposition: The wavelet representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(7):674–693.

Muller, D., Fleck, B., Dimitoglou, G., Caplins, B., Amadigwe, D., Ortiz, J., Wamsler, B., Alexanderian, A., Hughitt, V., and Ireland, J. (2009). Jhelioviewer: Visualizing large sets of solar images using jpeg 2000. *Computing in Science Engineering*, 11(5):38 –47.

NEMA (1989). Data compression Standard. Retrieved April 22, 2011, from the National Electrical Manufacturers Association ftp site: `ftp://medical.nema.org/medical/dicom/1989/PS2_1989.pdf`.

NEMA (2002). Digital Imaging and Communications in Medicine (DICOM) Supplement 61 : JPEG 2000 Transfer Syntaxes. Retrieved December 4, 2010, from the National Electrical Manufacturers Association ftp

site: `ftp://medical.nema.org/medical/dicom/2009/09_01pu.pdf`.

NEMA (2005). Digital Imaging and Communications in Medicine (DICOM) Supplement 105 : JPEG 2000 Part 2 Multi-component Transfer Syntaxes. Retrieved April 22, 2011, from the National Electrical Manufacturers Association ftp site: `ftp://medical.nema.org/medical/dicom/final/sup105_ft.pdf`.

NEMA (2006). Digital Imaging and Communications in Medicine (DICOM) Supplement 106: JPEG 2000 Interactive Protocol. Retrieved April 22, 2011, from the National Electrical Manufacturers Association ftp site: `ftp://medical.nema.org/medical/dicom/final/sup106_ft.pdf`.

NEMA (2009). Digital Imaging and Communications in Medicine (DICOM) Part 1 : Introduction and Overview. Retrieved April 22, 2011, from the National Electrical Manufacturers Association ftp site: `ftp://medical.nema.org/medical/dicom/2009/09_01pu.pdf`.

Noumeir, R. (2008). Integrating the healthcare enterprise process. *International Journal of Healthcare Technology and Management*, 9(2):167 – 80.

Noumeir, R. and Pambrun, J. (2010). Images within the electronic health record. *Image Processing (ICIP), 2009 16th*, pages 1761–1764.

Noumeir, R. and Renaud, B. (2010). Ihe cross-enterprise document sharing for imaging: interoperability testing software. *Source Code for Biology and Medicine*, 5(1):9.

Rabbani, M. and Joshi, R. (2002). An overview of the JPEG 2000 still image compression standard. *Signal processing: Image communication*, 17(1):3–48.

Schelkens, P., Skodras, A., and Ebrahimi, T. (2009). *The JPEG 2000 Suite*. Wiley-IS&T series in imaging science and technology. Wiley.

Subhendu, K. S., Shekhar, C., Kodali, S., Asati, A. R., and Gupta, A. (2009). Dual channel addition based FFT processor architecture for signal and image processing. *International Journal of High Performance Systems Architecture*, 2(1):35–45.

Taubman, D. and Marcellin, M. (2002). *JPEG2000: image compression fundamentals, standards, and practice*. Kluwer international series in engineering and computer science. Kluwer Academic Publishers.

Tuominen, V. J. and Isola, J. (2010). Linking whole-slide microscope images with DICOM by using JPEG2000 interactive protocol. *Journal of digital imaging : the official journal of the Society for Computer Applications in Radiology*, 23(4):454–62.

Zhang, J., Zhang, K., Yang, Y., Sun, J., Ling, T., Wang, G., Ling, Y., and Peng, D. (2011). Grid-based implementation of xds-i as part of image-enabled ehr for regional healthcare in shanghai. *International Journal of Computer Assisted Radiology and Surgery*, 6:273–284.