Endoscopic imaging results: web based solution with video diffusion

Joel Braga\textsuperscript{a,b*}, Isabel Laranjo\textsuperscript{a,b,c}, Domingos Assunção\textsuperscript{a}, Carla Rolanda\textsuperscript{b,d}, Luís Lopes\textsuperscript{e}, Jorge Correia-Pinto\textsuperscript{b,f}, Victor Alves\textsuperscript{a,c}

\textsuperscript{a}Department of Informatics, University of Minho, Braga, Portugal
\textsuperscript{b}Life and Health Sciences Research Institute (ICVS), School of Health Sciences, University of Minho, Braga, Portugal; ICVS/3B’s - PT Government Associate Laboratory, Braga/Guimarães, Portugal
\textsuperscript{c}CCTC - Computer Science and Technology Center, University of Minho, Braga, Portugal
\textsuperscript{d}Departments of Gastroenterology, Hospital de Braga, Portugal
\textsuperscript{e}Departments of Gastroenterology, Santa Luzia Hospital, Viana do Castelo, Portugal
\textsuperscript{f}Department of Pediatric Surgery, Hospital de Braga, Portugal

Abstract

In this paper we present a solution for diffusion of endoscopic imaging results. The aim is to solve the problem of endoscopy examinations access anywhere within a health care network by authorized health care professionals and researchers avoiding the need to repeat unpleasant exam procedures. The presented solution also enables streaming in real time of exams being carried out, which means that the health professionals can contribute with their expertise even when, for some reason, they cannot attend the exam onsite. All this information is archived and accessible at any moment. The proposed solution also provides tools to export the exams in any video format.

Keywords: Medical Video Streaming; e-Health; Endoscopy Video Acquisition; Telemedicine.

* Corresponding author.
E-mail address: joeltelesbraga@gmail.com.
1. Introduction

Therapeutic and diagnostic tests play a key role in health care delivery, helping the physician validate the diagnostic hypothesis [1]. Nowadays and with the technological evolution seen in the past years, the number of therapeutic and diagnostic tests prescribed/performed is growing increasingly [2]. Frequently the knowledge resulting from every medical practice and/or from therapeutic and diagnostic tests is not reused. This can be related to how the information is stored (e.g. paper, CD/DVD) or simply due to forgetfulness of its existence. These two reasons, occasionally contribute to an unnecessary repetition of therapeutic and diagnostic tests [3]. The lack of information sharing between different entities and health professionals can also be considered another reason for the needless repetition of the referred procedures [4].

Most of the IT infrastructures in health care entities do not allow any remote real time attendance by other professionals during the therapeutic and diagnostic exams due to the lack of specialized equipment. The lack of these functionalities means that all the involved professionals have to be present during the procedure, adding avoidable expenses.

There are many therapeutic and diagnostic tests (e.g. CT scan, X-Ray, Ultrasound, Endoscopy). Among them is the upper gastrointestinal endoscopy, which has low cost and good results. In this technique is used a thin and flexible tube, called endoscope, that has a light source and a very small video camera housed in its distal tip, thus allowing the visualization of the mucosa through the other end of the apparatus or in a video monitor[5], [6]. Beyond the visualization of different organs such as the esophagus, stomach and duodenum, the gastroenterologist can capture the most relevant images for his report. At the time of elaboration of the report it is not possible to review the exam or compare video segments with other procedures, which can lead to the repetition of the exam [7].

The presented scenario justifies the development of a solution, which suppresses these problems namely through real time or archiving and posterior deferred streaming of the video being acquired in the endoscopic exam. This streaming allows absent professionals to remotely assist and participate in the procedures. This will be achieved using a developed web-based application easily accessible anywhere as described in section 3.

2. Workflow using the proposed solution

The proposed solution allows a seamless integration in the current environment. It goes from the moment that the symptoms appear in the patient until the announcement of the result of the respective exam. The workflow of our solution can be seen in Fig. 1 with all the steps chronologically linked.

After the appearance of symptoms, the patient makes an appointment with his physician (Moment 1) [1]. In this appointment, any collected information is added to the patient’s clinical record. To validate the diagnosis the physician can prescribe one or more diagnostic tests, such as an endoscopy.

At the diagnostic test execution day (Moment 2), the gastroenterologist takes the necessary actions to perform the exam and to collect relevant information for the diagnosis process. In upper digestive endoscopy, the multimedia material from the exam could include videos and/or captured frames. During the procedures the video is visualized in the monitor of the procedure room. Using our solution it can be transmitted, in real time, to other entities, in order to allow absent physicians to give their opinion or simply to remotely follow the procedure.

After these two moments and in order to facilitate the sharing process, the video undergoes a set of processing steps, to obtain a shortened video with the most relevant information. However the physician can always have access to the original version, whenever he wants.
In a third moment (Report Writing), the gastroenterologist that performed the exam elaborates the corresponding report. While he proceeds with the report writing, he can access other functionalities present in the application, such as searching for similar cases or reviewing previous endoscopies of the same patient. After the report is completed, it is added to the patient’s clinical record.

At last, in moment 4 (Results Appointment), the physician who had prescribed the examinations receives the report and announces the results to the patient. Eventually the physician may not validate the diagnostic through the information originated by the prescribed examinations, so at any moment after the processing step, any health care professional can view the video remotely, provided he has permission.

Beyond the hospital context, this system also provides benefits in an academic / scientific context, since the videos can be anonymised and shared with investigation centers to work as training data, e.g. develop methods for automatic detection using previous cases.

3. MIVstream

Streaming is a mean to transmit real time multimedia information on a computer network. The main goal is to allow the receiver to preview the video in real time, while the data transfer is occurring. A server performs the video streaming remotely. Other way to deliver video over Internet is Progressive Download, where the video is stored in the user hard drive as the HTTP server sends it, which contrast with streaming since the video is not cached locally [8]. In our solution the streaming is used for sending video in real time and Progressive Download is used for past procedures videos.

The proposed web application MIVstream arises from the need to obtain video from endoscopy procedures previously archived or in real time, so it can be visualized and analyzed. The development of this system was
based on the VideoLan project that emerged in the academic environment towards the promotion and development of free and open source multimedia solutions. One of the tools developed by this organization is the VideoLan Client (VLC) [9]. This application consists in a very versatile player and a framework that allows a multiplicity of media inputs (e.g. File, DVD, CD, video acquisition, UDP, HTTP, FTP), format inputs (e.g. MPEG, AVI, MP4, MKV) and video formats (e.g. H.264, MPEG1/2/4, Theora). In addition, as a client, the VLC is compatible with the main platforms (e.g. Windows, Mac OS X, Linux) [9].

Using the libVLC library (an API written in the C language) it is possible to access the VLC full potential. The libVLC corresponds to the media player VLC application programing interface that allows the development of applications based on the VLC features. Thus, an abstraction of the most technical details like audio/video, coding/decoding and video transmission can be achieved.

In the MIVstream, the service provider is the health care entity that performs the endoscopic procedure, using a device called MIVbox, and the client is either, other MIVbox in the same facility or another health care facility via Web (e.g. Physician Office or Research Center) (Fig. 2)[7]. The Portuguese Health Network (RIS - Rede Informática da Saúde) connects health care facilities. This network is managed and secured by the Portuguese Ministry of Health so a secure channel for the transmission of data is guaranteed. When accessing from a Research Center or Physician Office the connection is made using a Virtual Private Network (VPN) connection.

![Diagram](image)

**Fig. 2** Streaming connection between the supplier and customers (adapted from [7])

*MIVbox* is the developed device that connects to the endoscope equipment and allows the archive of the procedures data (videos, frames, reports) and the corresponding streaming of video in real time.

The real time streaming of the procedure only occurs with the gastroenterologist’s previous authorization. Before the exam, the physician should indicate the health care entities or academic entities that it was given access for the visualization of the procedure. This authorization is granted through the definition of the
allowed Internet Protocol addresses or authorized user accounts (Fig. 3). This situation requires an authentication step where each user has to provide a given username and its corresponding access password. One of the most important aspects of the streaming process is the type of multimedia distribution over the network. There are two types of distribution that allows streaming communication: Unicast and Multicast. In the presented system there are several transmission profiles. In this process the physician can choose one profile either by using one pre-defined profile or by customizing the transmission parameters tailored to his needs.

The application responsible for the streaming procedure was written in the C programming language using the referred libVLC library. The streaming process starts with the creation of a libVLC instance \((\text{struct} \ \text{libvlc\_instance\_t})\) using the \text{libvlc\_new()} function (Fig. 4). This instance allows the creation of an abstract representation of a playable media \((\text{libvlc\_media\_t})\) using the \text{libvlc\_media\_new\_path()} function. This last function has as parameters a libVLC instance and a path that can be either a local filesystem path or a capture device path (e.g. in Linux “\text{v412://dev/video0}” corresponds to the webcam). To reproduce the video in the screen connected to the MIVbox, the creation of a libVLC media player \((\text{libvlc\_media\_player\_t})\) is enough. This is achieved by using the \text{libvlc\_media\_player\_new\_from\_media}, which receives a \text{libvlc\_media\_t} as parameter. The reproduction starts through the \text{libvlc\_media\_player\_play (libvlc\_media\_player\_t)} function. In addition, it is possible, for instance, to pause and stop the video, and obtain some information (e.g. FPS – Frames per Second), using the \text{libvlc\_media\_player\_pause}, \text{libvlc\_media\_player\_stop} and \text{libvlc\_media\_player\_get\_fps} functions.

Streaming can be obtained through the VLM (VideoLan Manager), which corresponds to the VLC core server. This server allows the streaming of media streams, and such is achieved by initializing the streaming with the \text{libvlc\_vlm\_add\_broadcast()} function that has the following parameters: libVLC instance, a name for the new broadcast and a path that can be either one of the previously named. This streaming can be started, paused and stopped through the \text{libvlc\_vlm\_play\_media()}, \text{libvlc\_vlm\_pause\_media()} and \text{libvlc\_vlm\_stop\_media()} functions. These functions receive as parameter the libvlc instance and the broadcast’s name. In the end, the process of freeing allocated resources is recommended, and this is achieved
using the `libvlc_vlm_release(libvlc_instance_t*)`, `libvlc_media_player_release(libvlc_media_player_t*)` and `libvlc_release(libvlc_instance_t*)` functions [9].

Fig. 4 libVLC features, functions and variables

The service client can access the streaming, in real time, and the past procedures, in deferred mode saved in the application server, through the MIVstream application (web application – Fig. 5), provided they are authorized to do so. At any moment physicians have access to their worklist for the current day and for the next days, always with the option to access patient’s information. The access to real time streaming is accomplished with the help of a plugin that allows the access to most of libVLC library functionalities and it does so through an API written in JavaScript [8][9]. This API allows playing all videos that the VLC media player can read on the browser.

An HTML page can reproduce these videos but there are differences between Operations Systems:

a) Mac OS X or Windows - Use the tag `<object>` with the attribute type="" application/x-fb-vlc” and a son tag `<param>` with the attributes name=""src"" and value =""[path to steam/file]”.

b) Linux - Use the tag `<embed>` with the attributes type=""application/x-vlc-plugin” and target=""[path to stream/file]”.

This is due to the different plugins that are used in the multiple operative systems [8] [9]. The video will be played in the `<embed>` and `<object>` elements, being evenly obligatory to define the width, height and id attributes. The first two are used to define the reproduction window size and the last one works as an identifier of the Document Object Model (DOM) object on which will be executed the javascript functions.

In a first phase we obtain a reference of the HTML element, with the previously given id, through the `document.getElementById` function. The plugin API functions will be executed on this variable, per instance: `.playlist.play()`, `.playlist.pause()`, `.playlist.stop()`, which allow to reproduce, pause and stop the video respectively. In the deferred videos case, it is also possible to manipulate the reproduction position and to obtain information, such as FPS, through the `.input.position`, `.input.rate` and `.input.fps` functions.

![Fig. 5 MIVstream Application Interface – Exam Visualisation](image)

It should be noted that while the streaming is carried out the video being captured by the MIVbox is stored to a video file (with parameters that can be set as mentioned above). Subsequently this video undergoes a number of processing steps that consist in feature extraction and removing non-relevant and repeated frames, remaining only the frames with clinical interest. The videos can then be chosen and viewed through a feature-based search for similar cases or with certain characteristics.

4. Discussion

Usually, in medical imaging, all interoperability between modality uses the Digital Imaging and Communications in Medicine (DICOM) Standard. This standard specifies a set of protocols, the syntax and semantics of commands and associated information (e.g. file formats and directory structure that must be followed) [12][13]. It assumes that the various devices are connected in a network and in conformance with the standard. We did not use this standard in our solution since endoscopes are not made in conformance with DICOM. In fact, endoscopy is not a part of the imaging department in health care institutions. The use of this standard implies the existence of a Service Class User and a Service Class Provider. They have to agree in a Service-Object Pair class that will be used, in order to be able to obtain a service by the provider. A solutions
presented in a study by Gunter et al. [14], videos need to be wrapped with a DICOM header containing necessary patient data and examination parameters and specific software must be installed in the desire devices so the application can be used.

Our solution uses a more classic client-server architecture where the client makes HTTP requests to the server and it responds in accordance with these requests. The underlying communication protocol agreement does not exist, simply consists of a client request and server response. The application interface is based in web technologies, thus a simple web browser can be used to access the system’s functionalities. This allows health professionals and researchers to avoid writing the exams on a DVD to visualize the video outside the places where these examinations are held.

The problem of access bandwidth in our system is resolved by an automated default choice of the most proper encapsulation methods and video codecs. According to Stephen and Alexandros [15] it is possible to dynamically adapt the rate of compressed video using a dynamic rate shaping and TCP congestion control technique. Our solution does not include the automation of the rate of compressed video, however this is to be considered for future work.

This solution introduces a streaming component that allows absent professionals to give their opinion remotely. Tests performed in a controlled and closed environment to the MIVstream application proved to be possible to perform streaming using a variety of encapsulation methods, video codecs, and resolutions. We must stress that streaming over the Internet has the effects of network jam and firewalls and as such the encapsulation method, video codec and resolution must be considered to obtain the best possible transmission quality without loss of information. And as such we must adapt the streaming to the speed and congestion of the network, allowing a dynamic change in the quality of streaming. To do that, it could be used a probe analysis to measure the quality of access, where the quality is measured by the bandwidth available.

Acknowledgements

This work is funded by ERDF - European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP-01-0202-FEDER-013853.

References


