

Performance of Remote Anatomy and Surgical Training Applications Under Varied Network Conditions

David Gutierrez, Amol Shah, and Dale A. Harris
SUMMIT / Department of Electrical Engineering
Stanford University, U. S. A.

degm@stanford.edu, amolshah@stanford.edu, and daharris@stanford.edu

Abstract: SUMMIT (Stanford University Medical Multimedia and Information Technologies) is currently developing hypermedia applications for remote anatomy and surgical training. These applications include interactive high-resolution 3-D imaging, 3-D streaming video and haptic (force feedback) tools to be used in both self and collaborative study modes. These applications are network-intensive and the perceived performance quality of each one changes in a different manner to varying network conditions. Part of the research conducted at SUMMIT has been focused on determining the network requirements for medical training applications like these. This paper first presents the results of experiments that study the usability of these applications as the network conditions change and then discusses the implications that these results have on the development of similar networked hypermedia educational applications and on the design of networks to be used for them.

Introduction

SUMMIT is currently developing 3-D applications and haptic tools that will be used for remote anatomy and surgical training. The 3-D imaging applications allow, for example, anatomy students to view high-resolution images, such as those of particular body structures, in 3-D with the use of special glasses. The 3-D streaming video applications allow the broadcast of surgical procedures. These applications are greatly enhanced by adding haptic tools to them, which provide the user with a force feedback mechanism that creates the illusion of a sense of touch on the images or models. Thus, the student actually receives tactile feedback through a computerized model of a particular organ or structure, or when practicing a surgical procedure.

The ultimate goal is to be able to run these applications over the Internet. Today's Internet, however, cannot meet the demands generated by this new generation of applications as it does not offer the networking functionality and Quality of Service (QoS) guarantees necessary. As a result, a good amount of research today, such as the Next Generation Internet (NGI) and Internet2 initiatives, is focused on developing an advanced Internet that can provide enhanced functionality. While most people think of these improved networks in terms of additional bandwidth, it is important to note that there are other parameters such as packet loss, delay and jitter, which significantly affect the performance of sophisticated applications such as these. Part of the ongoing research at SUMMIT is to study the effects these network parameters have on hypermedia applications. The remainder of this paper discusses the methodology and results from tests conducted at SUMMIT to determine what minimum network characteristics are required to support this kind of applications. These results are of importance both to the network planners and to the developers of future remote educational applications.

Background

The two applications studied in this paper are the *JPEG Server* (3D-imaging) and the *AutoHandshake* (haptics). The JPEG Server application provides interactive access to high-resolution stereo image sets. An example of such a set is the full 360° rotation views of a dissection through several layers of the hand (Fig. 1). This application allows students to perform a virtual dissection of the organ or structure being studied (such as hand, leg, skull), by removing different layers (such as skin, fat or tissue). It thus helps anatomy students to gain a better understanding of organs and structures at different layers and to create better mental models of them. The JPEG Server uses the TCP (*Transmission Control Protocol*) network transport protocol to establish the session, do initial setup procedures and transmit user requests. The actual image data is transmitted using a

modified UDP (*User Datagram Protocol*) that implements reliability. This reliability feature ensures that every received image is correct when displayed. Due to the high-resolution characteristic of the images, a high bandwidth network connection is required. Depending on the size and complexity of the image, the required bandwidth may go up to 40 Mbps (megabits per second).

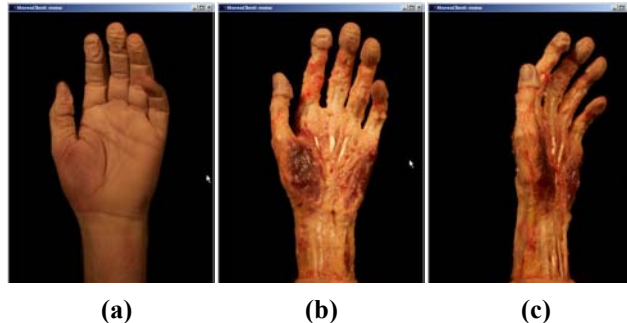


Figure 1: Sample 3D-imaging pictures: (a) Hand with all layers, (b) Hand with some layers (skin, fat, etc.) removed, (c) Same image as in (b) rotated.

The other application described here, the *AutoHandshake*, is a variant of the *Handshake* application, which is intended to train students remotely in surgical procedures. This is done by using a haptic device at each end and having the instructor guide the movements of the student remotely. The *AutoHandshake* variant is used when no instructor is available and the movements have been previously recorded on a simulator. This application uses the TCP protocol for initial setup in the same way as the JPEG Server application and then UDP to transmit the haptic data. For this particular application however, the UDP protocol is not modified to be reliable, since there is no time to correct network transmission errors while doing a movement. Unlike the JPEG server, the *AutoHandshake* is a low-bandwidth application that uses only up to 128 Kbps.

Method

The purpose of the experiments was to study the performance of the 3-D imaging and haptic applications under varying network conditions. Since it is difficult to actually test these applications on different networks in different geographic locations, it was necessary to recreate the network conditions in the laboratory. To do so, software tools were used to investigate different networks in order to get an idea of how much network parameters (throughput, packet loss, delay and jitter) differ among different networks. This provided ranges of values for the parameters of each of the networks examined. A network emulator was then used to recreate these conditions in the lab. Then, by connecting the applications across the emulator, the perceived quality of the applications was evaluated as the network parameters were varied.

Network Characterization

Network performance can be described quantitatively using four basic parameters. These are: (i) *throughput*, the available bandwidth, (ii) *packet loss*, how many IP (*Internet Protocol*) data packets are lost during transmission due to congestion, link failures or other problems, (iii) *delay*, the time it takes for a packet to get from one end to the other and (iv) *jitter*, the statistical variance of the delay. These parameters vary constantly, even on the same network. For example, depending on the route a packet takes to get from one point to another, it might need to go through a different number of routers along the way leading to different delays. Also, the amount of traffic that goes through a particular router changes constantly. Thus, the amount of packets queued waiting to be serviced at each router varies constantly, causing different per packet delays and, when the buffers are full, packet loss. Due to the best effort design of the IP protocol, packet loss is not unusual.

A commercial software package by NetIQ Corp called *Chariot* was used to characterize various network connections. *Chariot* operates as follows: small memory resident programs called *Performance Endpoints* are installed on computers that are located on the ends of the connection being evaluated. There is

also a controlling module residing on one of the endpoints or on a third-party computer. This module communicates with the endpoints and provides them with a script that simulates a particular network application, which can have a particular network transport protocol (TCP, UDP or others), data rate, session duration, packet size, etc. During the test, the endpoints measure, among other things, the average throughput, packet loss, delay and jitter and report these back to the controlling module.

For this experiment, network characterization was performed for four different types of two-endpoint network connections: a) connected across a 100 Mbps local area network, b) connected across a wireless local area network (WLAN) using the IEEE 802.11 standard, c) across an Internet connection with one endpoint at Stanford University and the other at the University of Wisconsin – LaCrosse, and d) across an Internet connection with one endpoint at Stanford University and the other at Sweden’s Royal Institute of Technology (KTH). The latter two connections go through Internet2 research and education infrastructures, which provide higher performing connections than across the commercial Internet. Table 1 shows the results that characterize the four networks using the throughputs and protocols used by the two applications under study.

	JPEG Server – 10 Mbps, Reliable UDP				AutoHandshake – 128 Kbps, UDP			
	WLAN	100 Mbps LAN	Wisconsin	Sweden	WLAN	100 Mbps LAN	Wisconsin	Sweden
Packet Loss (%)	N/A	0.000	0.006	0.130	0.000	0.000	0.000	0.024
Delay (ms)	N/A	0.200	29.500	82.500	1.750	0.200	29.500	82.500
Jitter (ms)	N/A	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 1: Network characterization results for different local and global connections

Network Emulation

Having obtained the values in Table 1, the next step was testing the perceived quality of each application under varying network conditions. To perform these tests using different values of packet loss, delay and jitter, a network emulator was used. The setup consisted of a gigabit fiber-optic connection between the server hosting the applications and the client computer. The network emulation occurred on a computer in the middle of the two endpoints. This computer used a public-domain network emulator software called NISTNet to manipulate the data traffic going between the two endpoints. NISTNet allows the emulator to behave like different networks by changing the values of packet loss, delay and jitter in a controlled fashion. Thus, using the values of the packet loss, delay and jitter obtained from the real networks, the emulator parameters were varied while running the applications to determine the boundaries of acceptable performance.

Application Evaluation

As mentioned earlier, the two applications studied were the *JPEG Server* (3-D imaging) and the *AutoHandshake* application (haptics). To determine the application quality, a conventional subjective grading scheme was used. The values for packet loss, delay and jitter were changed one at a time, and the users were asked to grade the performance of the application. It is worth noting here that the users were familiar with its performance under normal conditions. Each user graded their experience on a scale of 1 to 5. A score of 5 indicated “Excellent: network degradation is not noticeable”; a score of 4 indicated “Good: network degradation is noticeable, but not annoying”; a score of 3 indicated “Fair: network degradation is evident and annoying”; a score of 2 indicated “Poor: network degradation is severely affecting the user experience and the application is minimally useable”; and 1 indicated “Bad: network degradation has made the application unusable”. This grading scheme is similar to what has been extensively used previously for subjective video coding and transmission quality assessments (Webster et. al 1993, Watson, Sasse 1998, Jones, Atkinson 1998).

To test the JPEG Server, a set of 3-D images of the human hand was used (Fig. 1). Six test subjects familiar with the JPEG Server were asked to rate the performance of the application as the values of packet loss, delay and jitter were randomly changed. Similarly, for the AutoHandshake application, a subjective evaluation was performed using six test subjects that were familiar with the application and its capabilities under no network degradation. A Phantom Desktop haptic device, developed by SensAble Technology, was used during these tests. The AutoHandshake evaluation involved having the user make a certain movement with the Phantom device, and then having the server repeat the movement. The user would then determine if the accuracy and smoothness of the movement being repeated was equal to that of the original.

Results

Figures 2(a)-2(f) display the results for the JPEG Server and AutoHandshake under different packet loss, delay and jitter network conditions. In these figures, the characteristic parameter values for each of the four different network connections mentioned in Table 1 are indicated with markers (“100LAN”, “WLAN”, “U.Wisc” and “Sweden”). In Figures 2(a)-2(c), the “WLAN” marker is crossed out since this technology currently does not provide the necessary bandwidth for the JPEG Server application.

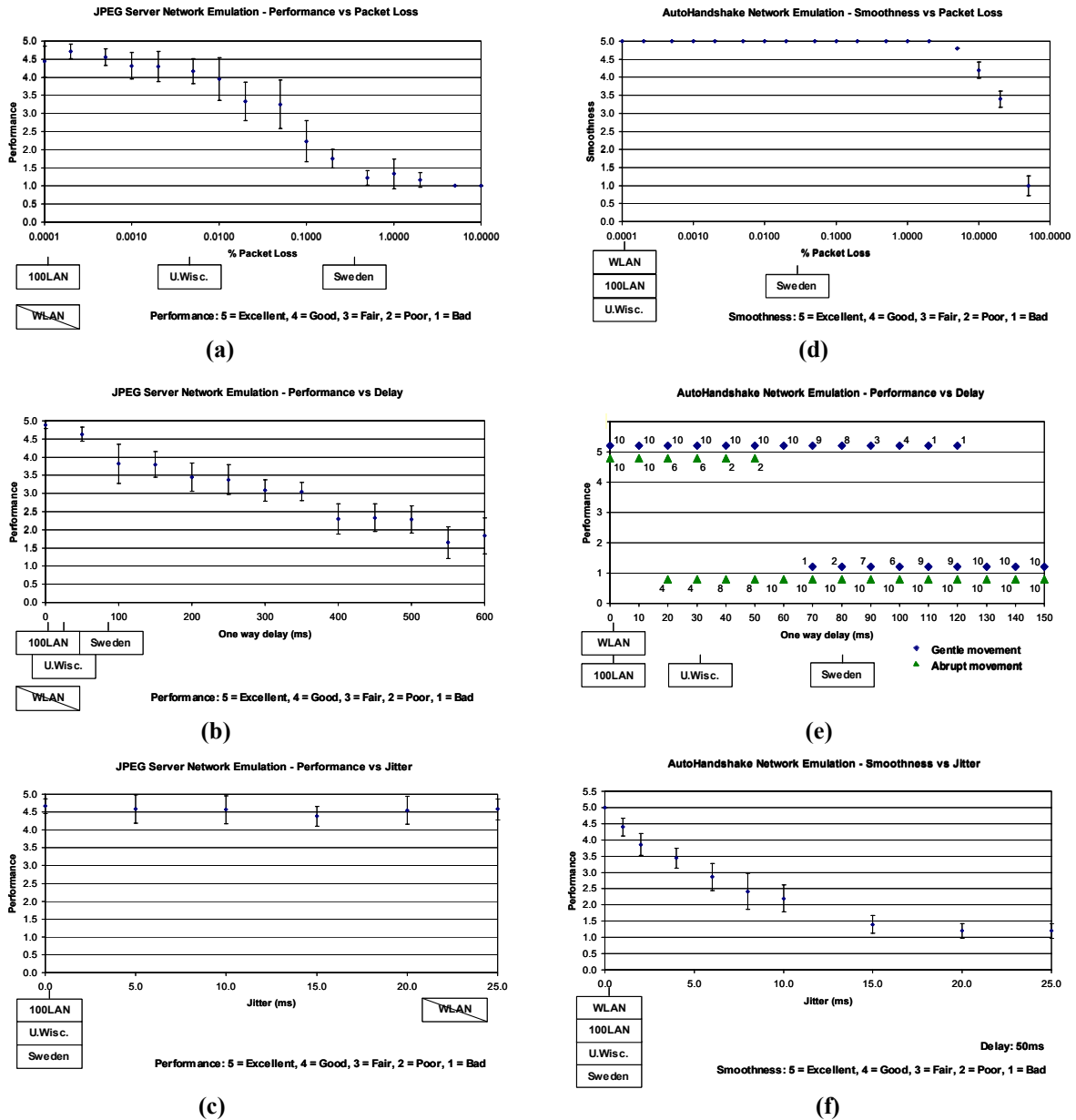


Figure 2: JPEG Server performance affected by (a) packet loss, (b) delay and (c) jitter. AutoHandshake performance affected by (d) packet loss, (e) delay and (f) jitter.

Figure 2(a) shows how the JPEG Server performance degrades with increasing packet loss, reaching “Bad” performance levels at about .8% packet loss. Since the application retransmits packets that are in error, packet loss causes the images to be delayed or not shown at all. Figure 2(b) illustrates how the performance also

degrades gradually with increasing delay, but never reaches the “Bad” level. It never reaches this level because the application will always eventually show the desired image, unlike the situation caused by packet loss, where it may never be shown due to continued retransmission requests until it fails. Figure 2(c) shows how the application is fairly insensitive to reasonable values of jitter. Jitter causes the JPEG server to have to wait just a few milliseconds more before displaying the image. This effect is barely noticed by the end user.

Figures 2(d)-2(f) show the results for the AutoHandshake application. Figure 2(d) shows that the application is very resilient to packet loss, starting to degrade only at values higher than 1%. Packet loss simply causes a few misses on points that describe the path that the haptic tool should be following. Since the sampling of points along the movement path is high, these missed points are averaged from the ones that do arrive, making the application fairly insensitive to packet loss. Figure 2(e) is different from the other ones, since the effect that network delay has on the application is different: it either works fine or it does not work at all. This is due to the force feedback loop in this application (Niemeyer, Slotine 1998). When the network parameters force the feedback loop into an unstable mode, the application effectively becomes unusable. In the studies, the application was run 10 times under a given delay and verified if it would work or not. The markers indicate the number of times the application did and did not become unstable. The results differ depending on whether the movement is gentle or abrupt. Again, this dependence occurs because of the dynamics of the force feedback

	JPEG Server	AutoHandshake
Bandwidth	40 Mbps	128 Kbps
Packet Loss	< .01 %	< 10 %
Delay (one way)	< 100 ms	< 20 ms (abrupt movements) < 80 ms (gentle movements)
Jitter	Not sensitive to jitter	< 1 ms

Table 2: Minimum QoS requirements per application

loop. The application performance is particularly sensitive to jitter, as Figure 2(f) illustrates, reaching “Bad” performance levels at around 15 ms of jitter. Jitter causes this degradation as the arrival of data points out of order causes the path to lose smoothness, which is easily perceived by the end user.

The graphs on Figures 2(a)-2(f) can be used to determine the minimum network requirements for these applications. Table 2 summarizes the requirements that these two applications impose on the network. These requirements have been taken as the values where the evaluated performance is in rated “good” or better. The threshold for the JPEG Server is at about 0.01% packet loss and 100 ms one way delay, as can be seen in Figures 2(a) and 2(b). For the AutoHandshake, the thresholds are at 10% packet loss and 1 ms jitter as Figures 2(d) and 2(f) illustrate. For its delay threshold, it depends on the movement: for abrupt movements it would be at around 20 ms, while for gentle movements it would be at around 80 ms, as Figure 2(e) illustrates. It is worth noting here that this evaluation has been made on a per parameter basis: that is, each parameter was treated independently. The results that different combinations of these parameters may bring have not yet been studied.

Table 2 illustrates one important point: simply providing more bandwidth is not sufficient to support these applications or others like them. For example, even though the AutoHandshake application needs only 128 Kbps to work properly, it has minimum delay and jitter requirements that may be hard to meet even for east coast to west coast connections within the U.S. This delay constraint for a haptic real-time application results in a limitation on the geographical distance allowable between a client and server or between two peers. This can become a serious problem since irrespective of the network functionality available, certain delay requirements cannot be met for geographically dispersed locations as the transmission speed is bound by the speed of light.

Discussion

The previous results illustrate how these two applications impose a series of requirements on the network being used. These results can be used along with the ones obtained for network characterization to estimate the performance of the applications on different real networks. Table 3 illustrates this. For example, the network connection from Stanford University to University of Wisconsin-La Crosse would support good to excellent performance of the JPEG Server application, but only inconsistent use of the AutoHandshake with abrupt movements. The connection from Stanford to KTH, Sweden would not be good enough for the JPEG Server due to packet loss, and delay requirements would not allow the normal operation of the AutoHandshake application. It is worth noting that both of these connections go through the Internet2 infrastructure, but even so,

the application requirements are higher than what they can provide. The only network connection of the ones characterized that can fully support both applications is the 100 Mbps LAN, which is local.

The delay requirement for haptic applications is particularly hard to handle from the point of view of the network implementers. First, there is a theoretical minimum delay, bounded by the speed of light. Then there is the delay that each router along the path causes during its normal operation due to queuing and switching. A QoS implementation that would allow minimizing the number of routers along a path or guaranteeing a maximum per router delay would be needed.

Connection Type	JPEG Server			AutoHandshake			
	Packet Loss	Delay	Jitter	Packet Loss	Delay		Jitter
					Gentle Movement	Abrupt Movement	
WLAN	N/A	N/A	N/A	Excellent	Excellent	Excellent	Excellent
100 Mbps LAN	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent
U.Wisc	Good	Excellent	Excellent	Excellent	Excellent	Excellent 60%; Bad 40%	Excellent
Sweden	Bad	Good	Excellent	Excellent	Excellent 80%; Bad 20%	Bad	Excellent

Table 3: Application performance in some characterized network connections

Conclusions

These experiments have shown how applications for remote medical training have stringent network requirements. Contrary to common belief, it is not just a matter of higher bandwidth. Other network parameters, such as packet loss, delay and jitter, are also very important. There is a need to better understand the limitations that current and near-future networks have (Tab. 1) and design the applications accordingly. The sensitivities and requirements that a particular application may have on a given network may be modified by adjusting a variety of design factors, including whether the application is real time or not, the kind of media that is being transmitted, the expectations of the end users and the network protocols being used.

A public network infrastructure that is expected to support remote medical training will need to provide QoS guarantees in terms of packet loss, delay and jitter (Tab. 2) and mechanisms that go beyond those provided today. Internet2 infrastructures today are not capable of appropriately handling these applications.

Some of these applications are currently in use at the Medical School, such as the JPEG Server for the course "Surg 219: Human Anatomy and Development". Future work at SUMMIT includes the evaluation of other network intensive applications, such as streaming 3-D video and peer-to-peer haptic applications, studying the network traffic patterns that are generated when the applications are in use and evaluating their scalability.

References

- Webster, A., Jones C., Pinson M., Voran S. and Wolf S. (1993). An objective video quality assessment system based on human perception. *Human Vision, Visual Processing, and Digital Display IV, 1993*, SPIE, San Jose, CA. 15-26.
- Watson A. and Sasse M. A. (1998). Measuring Perceived Quality of Speech and Video in Multimedia Conferencing Applications. *6th International Conference on Multimedia, 1998*, ACM, Bristol, England. 55-60.
- Jones, C. and Atkinson, D.J. (1998). Development of Opinion-Based Audiovisual Quality Models for Desktop Video-Teleconferencing, *6th International Workshop on Quality of Service, 1998*, IEEE, Napa, CA. 196-203.
- Niemeyer, G. and Slotine, J.-J. E. (1998). Towards Force-Reflecting Teleoperation Over the Internet. *International Conference on Robotics and Automation, 1998*, IEEE, Leuven, Belgium. 1909-1915.

Acknowledgements

This work was supported by a grant from the National Library of Medicine (NLM) awarded to SUMMIT to study the need for and define the characteristics of Next Generation Internet (NGI) technologies.