Proxy-based Visual Content Repurposing using Selection Algorithm

M. Shamim Hossain and Abdulmotaleb El Saddik
MCRLab, School of Information Technology and Engineering,
University of Ottawa, Ottawa, ON, K1N 6N5
{shamim, abed}@mcrlab.uottawa.ca)

Abstract – *This paper presents a proxy-based visual content repurposing approach, which is used to repurpose visual content dynamically for the transmission over heterogeneous networks and end devices. We use a series of repurposing proxies in a chain fashion between the server and multiple client devices. In order to find the appropriate chain of repurposing services that maximizes the user's satisfaction with the delivered content, we propose a novel service selection algorithm based on user's satisfaction with the quality of the repurposed content which is used as an optimization metric for the repurposing service selection and configuration process. Experimental results show that our approach performs complicated repurposing tasks by breaking them into a* series of intermediate tasks and executing them in distributed *repurposing proxies with minimal degradation of the final delivered content quality in terms of PSNR.*

Keywords – ***multimedia*** *content repurposing, user satisfaction, selection algorithm, performance measurement*

I. INTRODUCTION

Due to the recent advancement and diversity of video content coding standards, channel capacities, handheld device capabilities and users' preferences, there is a growing need for multimedia content repurposing. Content repurposing [1], also referred to as transcoding, is defined as the process of converting multimedia content from one form into another, then presenting it in such a way that solves the problem of a mismatch in the content coding standard, the device capability, the network and the user's characteristics.

We propose a selection algorithm, which finds the best repurposing rules to apply to the multimedia content where the user's satisfaction is used as an optimization metric for the repurposing service selection and configuration. Repurposing services, or transcoders, have been implemented through different proxies. Some of the repurposing services are developed in-house while others have been reused from publicly available reference software. These services repurpose the requested media content in a chain fashion, through different steps, such that the output format of one repurposing service is supplied to the input format of the next repurposing service until the requested content satisfies the end user's requirements and the outgoing target channel's requirements. Finally, repurposed content is delivered to the clients from the last repurposing service. This method facilitates complex repurposing tasks by combining one or more simple repurposing services in different steps. For instance, conversion of Motion JPEG (MJPEG) to H.264 might take place in two different steps using two different but simple repurposing services: MJPEG to H.263 [3] and H.263 to H.264/AVC [4] rather than performing it in a single step from MJPEG to H.264, which is computationally very complex. One of the benefits of our approach compared to other works e.g. [2] is that it enables for heterogeneous (between different coding standards) repurposing as well as homogeneous (within standard) repurposing. Additional proxies and/or services can also be easily added to the existing services to support any new media formats that need to be repurposed. Finally, experimental results show the quality of the repurposed content.

II. PROPOSED MULTIMEDIA CONTENT REPURPOSING SYSTEM ARCHITECTURE

Fig. 1 depicts the basic system architecture of our proposed proxy-based repurposing framework. As can be seen, different repurposing services can repurpose visual content for different users based on their preferences, channel bandwidth and device capabilities. To realize such architecture; we created several profiles that have descriptions of the capabilities and preferences of several clients, the services provided by each proxy and each repurposing service, and the transmission channel characteristics. While creating the profile, we considered several important issues [5] that include the following:

a) The repurposed content quality should be as high as possible or very close to the original encoded content.

b) The information in the original content (bit) stream should be reused in order to avoid deterioration.

c) The repurposing delay should be minimized in real time streaming.

d) If the target transmission channel conditions are unknown, the repurposing system should have the capability to dynamically convert the bit rate for the target channel.

As can be seen in Fig. 2, the initial connection between the client and the server is established using the Session Initiation Protocol (SIP) [6]. Initially the server sends a SIP "invite" message to the client containing the available ports and the IP of the server. The client replies with a 200 OK message to which it adds its profile including the device's capabilities, the IP-address of the device and the supported codecs. Upon connection with the server, the server stores the client's profile. The server then connects to each of its neighboring proxies. From each of the neighboring proxies, the server requests the profile of the proxy that includes one or more connected neighboring proxy and the repurposing services that are running on each neighboring proxy. Once all the information from all the proxies is received, the server generates a repurposing graph starting with the services that the server provides and expands the graph until all the proxies, and the required repurposing services have been taken into consideration and the client has been reached. Then the graph is simplified in order to ensure that no infinite loop exists and there is no backtracking. The graph is simplified in such a way that there is no path that does not lead to the receiver and there are no extra edges between the neighboring repurposing services. After the graph simplification, the server discovers the repurposing path that provides the best quality (user satisfaction) to the receiver. The path discovery uses the proposed satisfaction metric and selection algorithm presented in section 3.

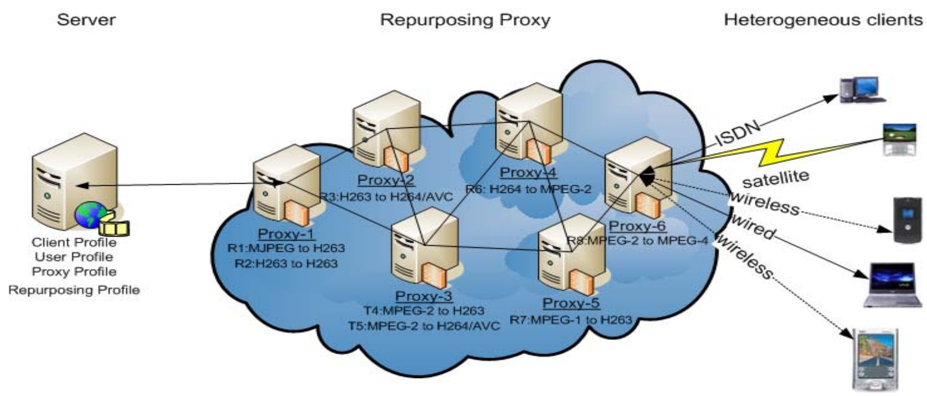


Figure 1: The basic architecture of the proxy-based visual content repurposing system

Once the best repurposing path is created by the server, it starts streaming the captured and/or pre-encoded visual content using the Real Time Protocol (RTP) [7]. The server then sends the path information and the stream to the first proxy in the chain. The proxy, upon receiving both path and stream, creates one or more repurposing services instantly. A stream can pass through multiple proxies and repurposing services and, if necessary, the stream can pass through the same proxy twice using two different repurposing services. However, it cannot use the same proxy twice in a row in the repurposing chain. The proxy then sends the rest of the path, as well as its own extracted information, to the next proxy in the chain such that the output formats of the previous repurposing service matches the input format of the next repurposing service in the chain. The last proxy in the chain simply forwards the results of its repurposed contents to the client. Once the streaming through a repurposing service stops, the reception port of the proxy is freed and the repurposing object is destroyed.

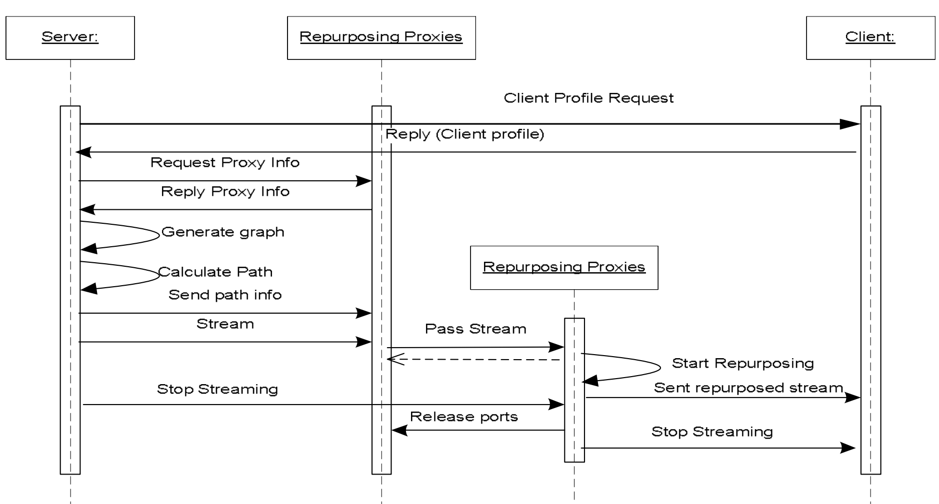


Figure 2: Realization of the system architecture with a sequence diagram

III. REPURPOSING SERVICE SELECTION ALGORITHM

The repurposing selection algorithm is one of the decision makers that find the best repurposing services to repurpose the content in order to make it acceptable to the user. The repurposing service selection algorithm uses the user satisfaction function as a metric to determine the best repurposing path. The satisfaction function is based on the multimedia application level variables such as the frame rate, resolution, coding distortion (quality), and bandwidth, as desired by the user. The satisfaction or preference of a user with each application level variables is expressed as a component satisfaction function si(yi), whose value is between 0 and 1. In this function yi is a multimedia application level variable.

For more than one multimedia application parameter, or variable, the overall satisfaction [8] of a user is determined as a combination function of the individual component satisfaction si, presented in Eq. (1) below:

$$S_{comb} = f_{comb}(s_i) = f_{comb}(s_1, s_2 .... s_m) = \frac{m}{\sum_{i=1}^{m} 1/s_i}, \qquad 0 \le s_i \le 1 \qquad ....(1)$$

Where, each satisfaction component s_1, s_2…, s_m is based either on frame rate, resolution, SNR quality or other parameters.

As mentioned in Eq. (1) the overall satisfaction function is low if one individual satisfaction function component is low. For instance, if a multimedia stream has a very high resolution but plays back at 3 fps, then the combination satisfaction is low.

In the algorithm described below, R_N is the set of repurposing services being considered by the system, i.e. the candidate set and R_V is the set of repurposing services that have already been considered. The candidate repurposing services set contain the repurposing services that have input edges coming from any repurposing services in the set R_V. At the beginning of the algorithm, the set R_V contains only the *Content_Sender* node, which contains the original media format; and R_N contains all the other repurposing services in the graph that are connected to the *Content_Sender*, and the *Receiver*. In each iteration, the algorithm selects the repurposing service R_i that generates the maximum user satisfaction. The user satisfaction is computed as an optimization function of the frame rate, the frame size or the quality for the output format of R_i, subject to the constraint of available bandwidth between R_i and its ancestor repurposing services. R_i is then added to R_V. The R_N set is then updated with all the connected neighbor repurposing services of R_i. The algorithm stops when the R_N set is empty, or when the *Receiver* node is added to R_V. The steps of the selection algorithm are given below:

1. Let $R_V \in$ {*Content_Sender*}.
   Let $R_N \in$ downstream neighbour {*Content_Sender*}.
   Let $\{R_i\}$ be the set of repurposing services.
2. If $R_N = \varnothing$ {i.e. no more repurposing service to consider and the receiver can not be reached through the repurposing path}, then TERMINATE (FAILURE).
3. For $\forall\, R_i \in R_N$, Compute the user satisfaction for all the repurposing services in R_N
4. Select the repurposing service R_i that has the maximum user's satisfaction
   $R_N = R_N - \{R_i\}$, $R_V = R_V \cup \{R_i\}$
5. If $R_i = Receiver$, then GOTO Step 8.
6. For $\forall\, R_j \in$ downstream neighbour $\{R_i\}$, $R_N = R_N \cup \{R_j\}$
7. GOTO Step 2.
8. Print repurposing path from the *Content_Sender* to R_i.

Let us illustrate the run time complexity of our algorithm. We assume, V and E are the nodes and the edges (links) respectively in a directed acyclic repurposing graph G(V, E). The satisfaction function is called once for each edge in the repurposing graph (steps 2-6), for a total of |E| calls. Steps 2-6 are repeated at most |V| times, as (repurposing node) vertex $v \in V$ is added at most once to the set R_N. Every time the repurposing services with the maximum user's satisfaction is selected and added to the repurposing service $\{R_i\}$. So, selecting the repurposing service with the maximum satisfaction takes O(V) time. Thus, the run time complexity of our algorithm is $O(|V|^2 + |E|)$.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

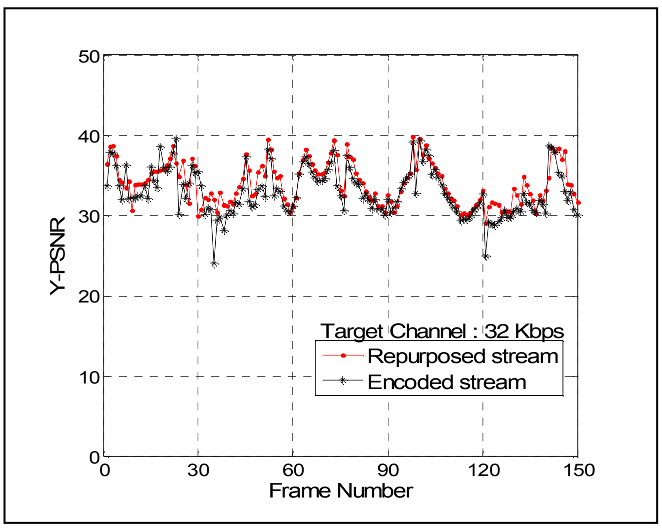
After implementation of the algorithm, we ported this algorithm to a multimedia conferencing service application. This application periodically scans for available multimedia repurposing services and determines the best available service, with the help of the proposed service selection algorithm, in order to render the multimedia stream to different clients based on the user's satisfaction. After running the application, we measured the quality of the repurposed content. For the measurement, we used a widely accepted objective measure of visual quality metric called the Peak Signal to-Noise-Ratio (PSNR) defined in Eq. (2). Where, MSE is the mean square error between the original content and the reconstructed visual content.

$$PSNR = 10\log_{10}\frac{255^2}{MSE}dB \qquad \text{.........(2)}$$

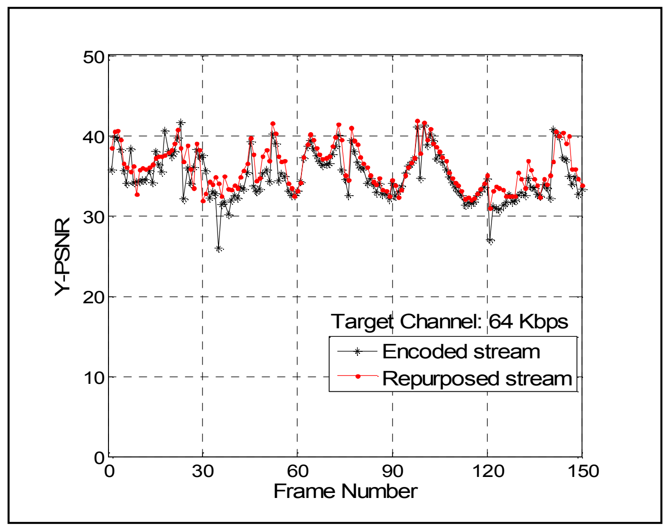
We reported some significant subsets of the results of the experiments conducted on our visual content repurposing system. One of the repurposing services that we developed was able to repurpose visual content from MJPEG to H.264 at different channel bandwidths. We measured visual quality by calculating and comparing the PSNR of repurposed visual contents and encoded the contents. As shown in Fig. 3, the visual content is repurposed from the M-JPEG (at 30 fps) to the H.264 (at 15 fps) for the target channel of 64 kbps and 32 kbps. The target channel bandwidth is dynamically identified by proxies and is sent to the server.

There is a sudden drop in both graphs in Fig.3 for the encoded data series, which is due to the changing of the scene. However, even when the scene changes happen at the 35th and the 121st frame, the visual quality of the repurposed frame is higher than that of the encoded stream because of the repurposing. There is a quality gain of 1db after repurposing visual content into H.264. This quality gain is due to the repurposing of the multimedia stream, where bandwidth is controlled.

We also found that running the repurposing service is a resource intensive task. A repurposing service uses about 30-40% of the processor (Pentium IV, 2.4 GHZ) depending on the motion of the video. This limits the number of repurposing services in a proxy to 2-3. A more powerful processor or a dual processor may be used to increase the number of repurposing services running on a proxy.



(a)



(b)

Figure 3: Quality comparison of repurposed stream and encoded stream at different bandwidth

V. CONCLUSION

In this article, we present a proxy-based visual content repurposing approach where proxies and repurposing services are distributed over heterogeneous networks. These repurposing services can be implemented from using in-house developed repurposing services and/or reusing publicly available reference coding software. One of the significant contributions of our approach is to introduce the usage of the repurposing path selection algorithm in order to select the best repurposing service in a dynamic way among several repurposing services. This algorithm creates the appropriate repurposing path based on the user's satisfaction. Experimental results show that our approach achieves excellent performance in terms of quality (PSNR) of the repurposed content for different target channels.

REFERENCES

[1] A. El Saddik and M.S.Hossain, "Multimedia Content Repurposing," in Encyclopedia of Multimedia, ed. B. Furht, Springer-Verlang, 2006.

[2] Z.M. Mao et. al,"Network Support for Mobile Multimedia using a Self adaptive Distributed Proxy," in Proc.NOSSDAV 2001, New York, June 25-26, 2001.

[3] ITU-T, "Video Coding for Low Bitrate Communication," ITU-T Rec. H.263, 2005.

[4] ITU-T and ISO/IEC JTC 1, "Advanced video coding for generic audiovisual services," ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, 2005.

[5] I. Ahmed, X. Wei, Y.Sun, and Y. Q. Zhang, "Video Transcoding: An Overview of Various Techniques and Research Issues," IEEE Trans. Multimedia, vol. 7, no. 5, pp. 793-804, 2005.
[6] J. Rosenberg et. al, "SIP: Session initiation protocol," IETF RFC 3261, 2002.
[7] RTP: A Transport Protocol for Real-Time Applications, IETF RFC 1889, 1996.
[8] A. Richards et. al. "Mapping User Level QoS from a Single Parameter", in Proc. MMNS'1998, Versailles, Nov. 1998.