STATISTICAL ANALYSIS AND MODELING OF HIGH DEFINITION VIDEO TRACES

Abdel Karim Al Tamimi, Raj Jain and Chakchai So-In
Dept. of Computer Science and Engineering, Washington University in St. Louis, USA
Email: {aa7, jain, cs5}@cse.wustl.edu

ABSTRACT

High definition video streams are gaining larger shares of the Internet usage for typical users on daily basis. This is an expected result of the current boom in the online standard and high definition (HD) video streaming services such as YouTube and Hulu. Because of these video streams' unique statistical characteristics and their high bandwidth requirements, they are considered to be a continuous challenge in both network scheduling and resource allocation fields. In this paper we provide a statistical analysis of over 50 high definition video traces that resembles wide varieties of high definition video traffic workloads. We performed both factor and cluster analysis on our collection of video traces to support a better understanding of video stream workload characteristics and their impact on network traffic. Additionally, we compare and evaluate different modeling approaches for high definition videos traces.

Keywords—Workload Characterization, Factor Analysis, Video Clustering, Multimedia Communications, Communication Networks.

1. INTRODUCTION

Web based video streaming websites open the doors to promising opportunities to distribute digital video contents to millions of people. Websites like YouTube [1] are now considered to be among the most daily accessed websites for Internet users. Such websites are now accounting for 27 percent of the Internet traffic, rising from 13 percent in one year [3]. This surge in traffic percentage can be explained by considering the latest surveys, as they show that the percentage of U.S. Internet users watching streaming videos have increased from 81% to 84.4%, and the average time spent per month increased from 8.3 to 10.8 hours/month in just three months [4,5]. Additionally, several websites have started to offer access to TV shows and selected movies, e.g. Hulu[2] and Netflix[6], which increased the reliance of the daily Internet users on such websites, and augmented their expectations of the level of services and quality of delivery. All these reasons inspire network researchers to put more emphasis on handling such demanding traffic.

Resource allocation and bandwidth control are dependent on their ability to predict and manage the demand of video streams. Therefore, the need of analyzing such challenging traffic, and possibly modeling it, is essential to allow better quality of service (QoS) support.

Modeling video streams is a challenging task because of the high variability of the video frame sizes. Such variability has been emphasized with the introduction of high definition codec MPEG4-Part10, also known as advanced video codec (AVC) and H.264. AVC codec provides better performance and compression rate (i.e. lower mean values) than their predecessors. Yet at the same time, they result in higher variability rates in frame sizes[7].

There have been several previous contributions that aimed to achieve a better understanding of the relationship between the behavior of the video traces and their impact on resource allocation. In [8], the authors presented a statistical and factor analysis study of 20 MPEG1 encoded video traces and the impact of such traffic on ATM networks. Similar approaches were presented in [9] with emphasis on video trace frame size distribution. The author in [10] performed a statistical analysis on four MPEG4-AVC encoded video traces with attention to the quantization effects on several statistical quantitative measurements and the correlation between video frames. In [11], the authors fitted one MPEG4-AVC encoded movie encoded with different quantization levels using Gamma density distribution function. In [7], the emphasis of the authors was to show the capabilities of the AVC standard versus its predecessor, viz., MPEG4-Part2.

In this paper, we present our work of analyzing and modeling over 50 HD video traces that we have selected from YouTube HD section. We aim through this contribution to investigate the main statistical characteristics that define a HD video trace. Such identifying process is important for two main reasons: it helps in clustering video traces depending on a certain statistical criterion to help choose the correct traffic workload, and for other data mining tasks. Additionally, it helps define the main statistical attributes of video traces that should be considered to achieve a valid statistical model. In our analysis, we also investigate the applicability of several video models in our pursuit for a general and a simple model that does not require significant statistical knowledge.

The rest of this paper is organized as follows: in the next section we discuss the methodology of selecting and encoding our collection of HD video traces. Section 3 illustrates the steps taken to perform both factor and cluster analysis on the video traces. Section 4 compares different

978-1-4244-7493-6/10/$26.00 ©2010 IEEE ICME 2010

Proceedings of IEEE International Conference on Multimedia anad Expo (ICME) 2010, Singapore, July 19-23, 2010

video traffic models and their validity to model HD video traces. Finally, section 5 concludes our paper.

2. COLLECTING AND ENCODING HD VIDEO TRACES

In order to provide a representative collection of video traces, we gathered over 50 YouTube HD videos [27]. The collected videos represent 15 different categories available to the site's users. Each category is represented in our collection with three videos. Additionally, we have chosen an additional set of videos to widen the selected levels of motion and texture.

To ensure common encoding settings for all the videos, we started by examining the videos encodings using *MediaInfo* [12] to determine the most common settings. Then, we confirmed our settings selection by refereeing to different encoding experts opinions [13]. Next, we converted the videos to their raw format (YUV420) and used JM reference software [14] to perform the encoding process using the common settings. The encoding procedure is both time and resources consuming process. The average time to encode one video is around 37 hours, and each video takes up to 4GB in space when it is converted to its raw format.

The selected common settings are shown in Table 1. We have chosen to use IDR (Instantaneous Decoding Refresh) frames since they allow better synchronization and seeking precision for the videos, which ultimately enhances the user's experience. The IDR period has been chosen to be 24, which is equal to the chosen frames rate, allowing seeking precision up to one second. We chose to use two B frames with three reference frames to follow the recommendations in [13]. The quantization parameters have been chosen to be close to the default settings for the JM encoder. Parameter *ProfileIDC* defines the video profile, which, in this case, is set to high. This parameter, along with the *LevelIDC* parameter specifies the capabilities that the client decoder must have in order to view the video stream correctly. Parameter *NumberBFrames* specifies the number of B slices or frames between I, IDR and P frames. All the encoded video traces and their autocorrelation function (ACF) plot, partial autocorrelation function (PACF) plot, cumulative distribution function (CDF) plot, and their encoding process statistics are all available as a part of this contribution to the research community with all the used tools and scripts [27].

Table 1. Encoding Parameters for the Selected YouTube Video Collection

Encoding Parameter	Value
FrameRate	24
Width	1280
Height	720
ProfileIDC	100 (High)
LevelIDC	40 (62914560 samples/sec)
NumberBFrames	2
IDRPeriod	24
NumberReferenceFrames	3
Quantization Parameters (QP)	I=28, P=28,B=30

3. FACTOR AND CLUSTER ANALYSIS OF VIDEO TRACES

In this section we discuss the steps taken to perform a full statistical analysis of the collected video traces in order to achieve a better understanding of the main factors that can be used to represent a video trace in order to develop a representative statistical model.

Multivariate analysis is used to reveal the full structure of the collected data, and any hidden patterns and key features [15]. Multivariate analysis is used especially when the variables are closely related to each other, and there is a need to understand the relationship between them. We have computed the following statistical quantitative values for traces frame sizes: mean, minimum, maximum, range, variance, standard deviation, the coefficient of variance, and the median value. In addition, we computed the Hurst exponent value, as shown in equation 1, which indicates the video sequence's ability to regress to the mean value, with higher values indicating a smoother trend, less volatility, and less roughness. Its value varies between 0 and 1. This is also an indication of the long range dependence (LRD) between the frames.

$$S = \sum_{i=1}^{N} (x_i - \bar{x}), \quad R = \frac{\max(S) - \min(S)}{\sqrt{\frac{S^2}{N}}}$$

$$Hurst\ Exponent = \frac{\log(R)}{\log(N/2)} \quad (1)$$

where x_i is the frame size at index i, $\bar{x}$ is the mean frame size, and N is the number of frames in the trace. We also computed the *skewness* value that represents the symmetry of the observed distribution around its center point as illustrated in equation 2.

$$Skewness = \frac{\sum_{i=1}^{N} (x_i - \bar{x})^3}{(N-1)\cdot(std)^3} \quad (2)$$

here *std* is the standard deviation of the frames sizes. Additionally, we computed the *kurtosis* value, which is an indication whether the observed distribution is peaked or flat relative to a normal distribution. The kurtosis equation is illustrated below:

$$Kurtosis = \frac{\sum_{i=1}^{N} (x_i - \bar{x})^4}{(N-1)\cdot(std)^4} \quad (3)$$

As Table 2 shows, the collected videos represent a statistically diverse data samples. As mentioned before, the video frame sizes variance is considerably substantial. The table shows the most important variables that have been collected. We noticed through our preparation analysis that *min* variable does not contribute to the total variance significantly, and thus it was disregarded. Both *max* and *range,* and *variance* and *standard deviation* pairs are almost identical. We picked *range* and *variance* to represents the two pairs respectively. In the next subsections, we will discuss the methodology and results of performing both factor and cluster analysis.

Table 2. Range of Statistical Values for the Collected Video Traces

	Mean	Range	Variance	Hurst	Coefficient of Variance	Median	Skewness	Kurtosis
Max	83340.43	1198416	13767760363	0.902836	3.9860815	62748	6.58066	61.34631
Min	9782.01	65576	154362485	0.498937	0.6875022	448	0.2287191	-1.643709

3.1 Principal Component Analysis

One of the most common factor analysis methods is principal component analysis (PCA), where a group of possibly related variables are analyzed and then reduced to a smaller number of uncorrelated *factors*. These factors are linear combinations of the analyzed variables. By performing this process, we aim also to minimize the number of variables to represent a video trace [15].

Table 3. Correlation Between The Selected Variables

	Mean	Range	Var	Hurst	c.var	Median	Skew	kurt
Mean	1	0.48	0.73	0.48	-0.40	-0.9	-0.36	-0.23
Range	0.48	1	0.74	0.34	0.19	0.25	0.51	0.6
Var	0.73	0.74	1	0.36	0.13	0.41	0.13	0.14
Hurst	0.48	0.34	0.36	1	-0.44	0.41	0.25	0.17
c.var	-0.40	0.19	0.13	-0.44	1	-0.56	0.71	0.51
Median	-0.9	0.25	0.41	0.41	-0.56	1	-0.49	-0.33
Skew	-0.36	0.51	0.13	0.25	0.71	-0.49	1	0.93
Kurt	-0.23	0.6	0.14	0.17	0.51	-0.33	0.93	1

Using factor analysis we obtained the smallest number of variables to represent each video trace. Table 3 shows the correlation between the selected variables. These factors collectively represent the majority of the samples variance. The importance of each factor is represented by its *eigenvalue*. To determine the number of factors to extract we used *Kaiser-Guttman* rule [16]. We excluded the factors with eigenvalue less than 1. We supported our selection by performing the *Scree* test [17], where we plotted the relationship between the number of factors and their cumulative contribution to the total variance of the data set, and we looked for either large spaces between the plotted variables or a *knee* in the graph to determine the number of factors. Our analysis resulted in choosing two factors with the following eigenvalues: $\lambda_1 = 3.51$, and $\lambda_2 = 2.82$. These factors accounts for 79% $[(\lambda_1 + \lambda_2) / 8]$ of the total standardized variance. We assured that the number of factors is sufficient to explain the inter-correlations among variables by performing several non-graphical tests [18]. To simplify the factor structure and spread out the correlations between the variables and the factors (their *loadings* values) as much as possible, we performed both orthogonal and oblique rotations on the factors [19]. We chose *varimax* orthogonal rotation as it gave the best results. As shown in Fig. 1 the two significant groups are the mean and skewness groups. Table 4 shows the loadings values for both *varimax* rotated and un-rotated factors.

Table 4. Estimated and Rotated Factors Loadings

	Estimated		*Rotated (varimax)*	
Variable	**F_1**	**F_2**	**F_1***	**F_2***
Mean	0.84	0.46	**0.93**	-
Range	-	0.95	0.73	0.62
Variance	0.39	0.80	**0.84**	-
Hurst	0.62	-	0.64	-
C. Var	-0.75	0.35	-	0.77
Median	0.87	-	0.77	-0.46
Skewness	-0.75	0.62	-	**0.97**
Kurtosis	-0.62	0.67	-	**0.91**

As can be noticed, the rotated factors are better spread out and simpler to interpret. From Table 4 we can note that the first factor F1* defines mainly *mean* and *variance* values. The second factor defines mainly *skewness* and *kurtosis* values. We chose *mean* to represent the first factor since it has the highest load. We chose kurtosis as a representative of F2* since it has the lowest correlation between it and the *mean* (-0.23). This analysis shows the importance of *skewness* and *kurtosis* in HD videos traces. These variables were considered irrelevant in previous video analysis [6]. This realization can be explained by the dependence of these variables on the standard deviation that accounts for a significant proportion of the total variance of HD videos traces.

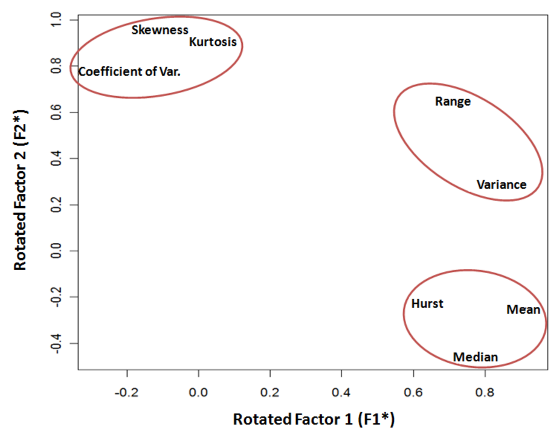


Fig. 1. Scatter Plot of Varimax Rotated Factors F_1* and F_2* in the Space of the Two Principal Components

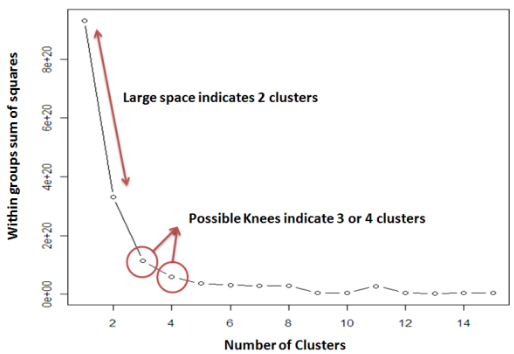
3.2 K-means Clustering

We have demonstrated that the selected two factors are sufficient to characterize the movie traces. The second step of our analysis is to group the collected video traces into clusters. We used one of the most popular clustering methods: *k-means* clustering algorithm [20]. K-means algorithm achieves clustering by minimizing the within-cluster sum of squares as shown in equation 4:

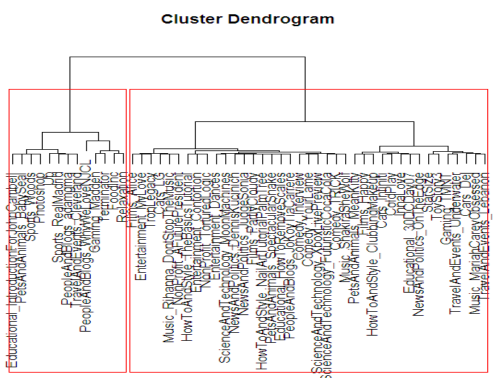
$$\arg\min_{s} \sum_{i=1}^{k} \sum_{x_j \in S_i} \left\| x_j - \mu_i \right\|^2 \quad (4)$$

where x_i is the video trace at index i, k is the number of sets ($k<n$, n:number of video traces), S_i is the i-th set, and μ_i is the mean of S_i.

Our next step is to estimate the number of clusters or groups to consider for k-means clustering. PCA helps give an insight of how many clusters the data samples can be grouped into [21]. In our case, PCA suggests that we need two clusters. In order to verify the analysis results from PCA, we proceeded with computing the within-cluster sum of squares for different number of clusters. Our aim is to select the minimum number of clusters that allow the minimal possible value for the within-cluster sum of squares. By plotting these values to represent a graph similar to the *scree* test, the large spaces between the plotted variables and the graph possible knee indicates the possible values: two, three, and four clusters as shown in the Fig. 2 (a). To further investigate the best possible number of clusters to use, we performed a hierarchal clustering to identify the number of clusters using *Ward*'s method [20]. As shown in Fig. 2(b), the video traces are divided into two main clusters. Grouping into two clusters choice was verified by performing *silhouette* validation method [22].



(a) Within Groups Sum of Squares vs. Number of Clusters



(b) Hierarchal Clustering Result

Fig. 2. Determining Number of Clusters using *Scree* Test and Hieratical Analysis

By performing k-means clustering we grouped the video traces into 2 clusters. Table 5 shows the two chosen principal components corresponding to the centriods of the two clusters, and the two clusters main members. Figure 3 shows the distribution of video groups over the two clusters.

Table 5. Clustering Results Using K-Means Clustering

Variables	Cluster 1	Cluster 2
Mean	59251	32582
Kurtosis	12.028829	9.099512
No .of Elements	13	39
Main Video Groups	Films, People and Blogs, Sports, Educational	Films, Music, News, Comedy, Cars

In summary, video traces that belong to cluster 2 have significantly lower mean values, and have considerably low peaks compared to normal distribution, and lighter tails as indicated by their low Kurtosis values.

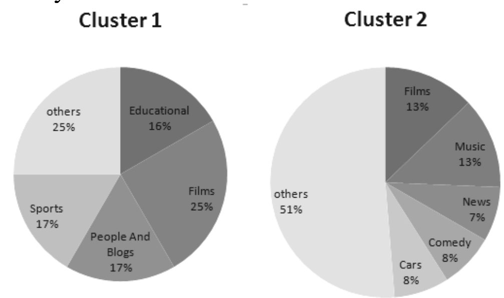


Fig. 3. Distribution of Movie Groups over The Two Clusters

In addition, we notice the following: films category video traces are spread across both clusters. Most blogs and sport category videos are characterized as *peaky* video traces because of their content; news and comedy videos are less peaky and have lower means than other movies.

In this section, we demonstrated our results of performing both factor and cluster analysis on our collection of video traces. Both methods of analysis give us a better understanding of the distribution of the movie traces and the key statistical attributes.

4. MODELING HD VIDEO TRACES

In this section, we discuss and compare three statistical models to represent HD video traces. Several models to represent VBR (Variable Bit Rate) MPEG traffic have been proposed in the last decade. Some of the models proposed are based on Markov chain models, which are known for their inefficiency in representing the LRD characteristics of MPEG traffic [28, 29]. Due to the high influence of LRD, multiplicative processes have been considered like Fractional ARIMA (FARIMA) which have been shown to be accurate, computationally demanding and provide marginal improvements over ARIMA[30]. Our aim is to select a simple to implement, accurate and applicable model for all video traces without the need of significant statistical background. The chosen model should not require video-specific complex and tedious steps. The model should be able to not only represent video frame size distribution, but also the correlation between the frames. These attributes are important to achieve the desired results and to allow the analysis of the 52 video traces. We picked three modeling methods: autoregressive (AR) model, autoregressive integrated moving average (ARIMA) model [24], and simplified seasonal ARIMA model (SAM) [23]. All these models consider Akaike's Information Criterion (AIC) as their optimization goal. AIC is defined as:

$$AIC = 2k + n[\ln(RSS / n)] \qquad (5)$$

here k is the number of parameters, n is the number of the video frames, and RSS is the residual sum of squares. AIC defines the goodness of fit for the models, considering both their accuracy and complexity defined by their number of parameters. Lower AIC values indicate better models in terms of their validity and simplicity. Below is a short description of each modeling method.

4.1 AR Modeling with Maximum Likelihood Estimation

Autoregressive fitting takes into consideration the previous values of the fitted trace. An autoregressive model of order p can be written as:

$$X_t = \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \tag{6}$$

where φ_i is the i-th model parameter, and ε_t is white noise. We use maximum likelihood estimation (MLE) to estimate the model parameters of the AR model. Using AR to fit the video traces is considerably simple process, but it does not always yield accurate results. Additionally, each video traces has its own set of parameters in terms of their numbers and their values.

4.2 ARIMA Modeling

Autoregressive integrated moving average model is a mathematical class model with both autoregressive and moving average terms. Moving average (MA) terms describe the correlation between the current value of the trace with the previous error terms. The integrated or differencing part of the model can be used to remove the non-stationarity of the trace. ARIMA is usually referred as ARIMA(p,d,q) where p is the order of the autoregressive part, q is the order of the moving average part, and d is the order of differencing. ARIMA model can be written as:

$$\left(1 - \sum_{i=1}^{p} \phi_i L^i\right)(1-L)^d X_t = \left(1 + \sum_{i=1}^{q} \theta_i L^i\right)\varepsilon_t \tag{7}$$

where L is the lag operator, and θ_i is the i-th moving average parameter. We used *auto.arima* statistical method from *forecast* package [24], which implements a unified approach to specify the model parameters. This approach also takes into consideration the seasonality of the trace. This approach also results in a separate set of parameters for each video trace in terms of their numbers and their values.

4.3 SAM Model

SAM is a mathematical model based on Seasonal ARIMA (SARIMA) models [23]. SARIMA models aim to achieve better modeling by identifying both non-seasonal and seasonal parts of data traces. SARIMA is described as:

$$SARIMA = (p,d,q)\times(P,D,Q)^s \tag{8}$$

where P is the order of the seasonal autoregressive part, Q is the order of the seasonal moving average part, D is the order of seasonal differencing, and s denotes the seasonality of the series. SAM as SARIMA model can be written:

$$SAM = (1,0,1)\times(1,1,1)^z \tag{9}$$

where z is the autocorrelation function (ACF) lag seasonality, in our case this is equal to the frames rate. SAM provides a unified approach to model video traces encoded with different video codec standards using different encoding settings [25]. Although the model is presented to model mobile video traces, we investigate its ability to model HD video traces with higher resolutions. SARIMA models require a certain degree of analysis to identify the model parameters number and their values. SAM, on the other hand, has only four parameters, and therefore each model is represented with only four values. The values the parameters are determined using *Nelder-Mead* method [26]. The four parameters are: autoregressive, moving average, seasonal autoregressive, and seasonal moving average. Therefore, using SAM simplifies the analysis process that is usually required for seasonal series.

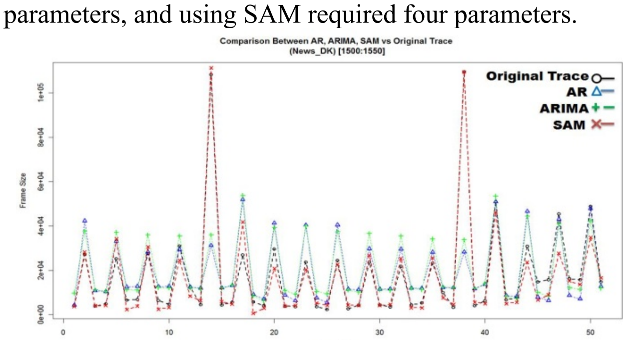
4.4 Modeling Results

After performing the modeling analysis on our collection of 52 HD video traces, we evaluate the achieved results by simply comparing the sum of the AIC values for all the modeled video traces. We also calculated the number in which each model has scored the best AIC for a certain video trace. The results are shown in Table 6. It can be noted that SAM achieved the best AIC results, while AR and ARIMA came in second and last place respectively.

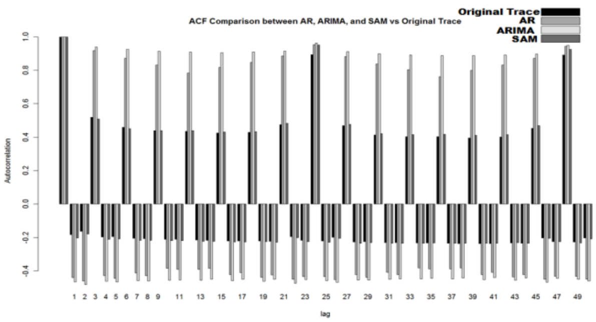
Table 6. Comparison between AR, ARIMA and SAM using AIC

	AR	ARIMA	SAM
Total AIC	3473929	3492401	3344490
No. of Best AIC	6	3	43

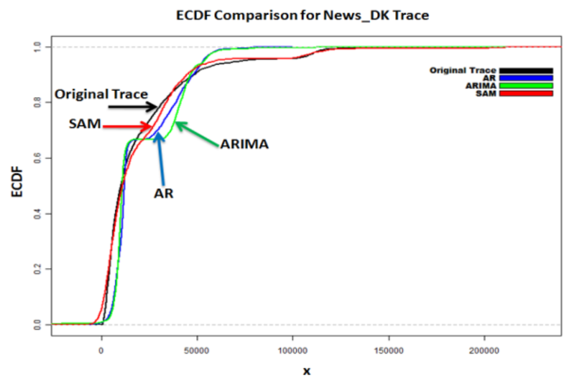
Additionally, we performed several graphical comparisons for all the video traces by comparing the original video traces, their auto correlation function (ACF) plots, and their empirical cumulative distribution function (ECDF) plots to ones achieved by the different models. Figure 4 shows an example of one of the compared video traces. As we notice, SAM has better results and represents the traces statistical characteristics better than the other two models. For this example, modeling using AR required 12 parameters, using ARIMA required two AR parameters and five MA parameters, and using SAM required four parameters.



(a) Trace Comparison (frames between 1500-1600)



(b) ACF Comparison (first 50 lags)



(c) ECDF Comparison

Fig. 4. Modeling Comparisons for AR, ARIMA, and SAM

All graphical comparison results for the HD video traces are also available through our website [27] as a part of this contribution.

5. CONCLUSIONS

In this paper, we presented our results of analyzing more than 50 HD video traces using both factor and cluster analysis. We showed that skewness and kurtosis statistical variables are principal components to HD video traces. We illustrated and compared using both AIC values and graphical comparisons the capabilities of three mathematical models. SAM's capability to accurately model HD video traces has been proven, and it provided better results than AR and ARIMA models. Finally, as a part of this contribution: all the video traces, tools, and graphical comparisons are made available to the research community.

6. REFERENCES

[1] YouTube Website, URL=[http://www.youtube.com].

[2] Hulu Website, URL=[http://www.hulu.com].

[3] C. Albrecht, "Survey: Online Video Up to 27% of Internet Traffic", URL= [http://tinyurl.com/yzpzoew].(March 10)

[4] Comscore Press Release in Aug.2009, URL=[http://tinyurl.com/l4o3rs]

[5] Comscore Press Release in Nov.2009, URL=[http://news.websitegear.com/view/149267](March 10)

[6] Netflix website, DVD Rental and High definition video streaming service,URL=[http://www.netflix.com].(March 10)

[7] G. Van der Auwera, P. David, M. Reisslein, "Traffic characteristics of H.264/AVC variable bit rate video, " IEEE Comm. Magazine, Volume 46 Issue 11, pp 164-174.

[8] P. Manzoni, P. Cremonesi, G. Serazzi, "Workload Models of VBR Video Traffic and Their Use in Resource Allocation Policies," IEEE/ACM Transactions on Networking (TON), Volume 7,Issue 3 (June1999),pp.387-397

[9] O. Rose, "Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems," 20th Conference on Local Computer Networks, 1995.,Volume, Issue , 16-19 Oct 1995 pp.397-406

[10] Lanfranchi, I. Laetitia,"MPEG-4 AVC traffic analysis and bandwidth prediction for broadband cable networks", MS Thesis, 2008, Georgia Tech.

[11] H. Koumaras, C. Skianis, G. Gardikis, A. Kourtis, "Analysis of H.264 Video Encoded Traffic", Proccedings of the 5th International Network Conference (INC2005), S.M. Furnell, P.S. Dowland, G. Kormentzas (eds.), Samos, pp. 441-448.

[12] MediaInfo, URL=[http://mediainfo.sourceforge.net/en]

[13] Jan Ozer, "Producing H.264 Video for Flash: An Overview", URL= http://tinyurl.com/yj979sy

[14] JM Ref. Software, URL=http://iphome.hhi.de/suehring/tml/

[15] Brian S. Everitt , "An R and S-Plus® Companion to Multivariate Analysis," Springer 2007.

[16] H T Kaiser, "The Application of Electronic Computers to factor analysis", Educ. Psychol. Meas. 20:141-51, 1960.

[17] R. B. Cattel, "The scree test for the number of factors," Multivariate Behav. Res. 1:245-76, 1966.

[18] G. Raîche, M. Riopel, J. Blais, "Non Graphical solutions for the cattell's scree test", Annual meeting of the Psychometric Society, Montréal, Canada.

[19] G. Kootstra, "Project on exploratory Factor Analysis applied to foreign language learning," 2004.

[20] M. Norusis, "SPSS 17.0 Statistical Procedures Companion," Prentice Hall, 2009.

[21] C. Ding, X. He, "K-means clustering via principal component analysis," ACM Inter. Conf. Proceeding Series; Vol. 69, 2004

[22] Cluster Validity Algorithms,URL=http://tinyurl.com/yj8jz9w, fetched on oct-2009.

[23] Al Tamimi, A., Jain, R., and So-In, C., "Modeling and generation of AVC and SVC-TS mobile video traces for broadband access networks," In Proceedings of the First Annual ACM SIGMM Conference on Multimedia Systems. MMSys '10.

[24] R. Hyndman, Y. Khandakar, "Automatic Time Series Forecasting: The forecast Package for R", Journal of Statistical Software, Vol. 27, Issue 3, Jul 2008

[25] A. Al Tamimi, C. So-In, R. Jain, "Modeling and Resource Allocation for Mobile Video over WiMAX Broadband Wireless Networks," IEEE Journal on Selected Areas in Comm., Vol. 28 Issue 3, pp. 354-365, April 2010.

[26] J. A. Nelder, R. Mead "A simplex algorithm for function minimization," *Computer Journal* 7, pp. 308–313.1965

[27] Reference not specified to allow blind refereeing required by the conference. Will be specified in the published version.

[28] A.M. Dawood, M. Ghanbari, "Content-Based MPEG Video Traffic Modeling," IEEE Transactions on Multimedia, Volume 1, Issue 1, Mar 1999, pp.77– 87.

[29] Y. Sun, J.N. Daigle, "A Source Model of Video Traffic Based on Full-Length VBR MPEG4 Video Traces," IEEE Global Telecomm. Con., 2005. GLOBECOM'05,Vol. 2, 28, 5 pp.

[30] O. Lazaro, D. Girma, J. Dunlop, "A Study of Video Source Modeling for 3G Mobile Communication Systems," Proc. of 1st Int. Con. on 3G Mobile Comm. Tech., 2000, Conf. Publ. No. 471, pp.461 – 465.