# Social Media Video Content Diversity Visualization

KLIMIS NTALIANIS[1] and NIKOLAOS MASTORAKIS[2]

[1]Department of Marketing
Athens University of Applied Sciences (TEI of Athens)
Agiou Spyridonos, Egaleo, Athens
GREECE
kntal@teiath.gr

[2]Industrial Engineering Department
Technical University of Sofia,
Sofia,
BULGARIA
mastor@tu-sofia.bg

*Abstract:* - In this paper diversity visualization of video content posted on Social Media is efficiently performed, by proposing an unsupervised intelligent video analysis scheme. The proposed scheme assumes several different videos, posted by several different social media users. Its aim is to provide an overall compact view of the diverse video content people share. Similarly, it is like providing a summary of the total posted visual information (for a specific time instance or interval), so that users can take an idea of what is happening outside their micro-world. Towards this direction, each video is analyzed and key-frames are extracted based on a correlation measure and a social computing algorithm. The final summary is created by extracting the most uncorrelated frames among all key-frames, so that the diversity of the visualized content is kept. Experimental results are presented, to denote the full potential of the proposed scheme, its advantages as well as important issues for future work.

*Key-Words:* - Social Media, Social Computing, Video Summarization, Content Diversity, Correlation

## 1 Introduction

Social networking has become a "global phenomenon". According to statista.com [1], the number of users for 2016 is estimated to 2.34 billion! Currently social media have been well established as a category of online discourse where people create content, share, bookmark and network at an unprecedented rate. Because of their ease of use, speed and reach, social media are fast changing the public discourse in society and setting trends and agendas in topics that range from the environment and politics to technology and the entertainment industry.

However one of the major problems of social media is the fact that they are really chaotic. Billions of users post billions of items every day [2] and users see on their timelines, only some of the videos posted by their friends As a result, currently it is impossible to follow the total activity (or even a small portion of it), even if someone focuses on just one of the social media. This may mean lost opportunities, extremely limited informing, confined potential etc.

By taking into consideration the aforementioned deficiencies, in this paper we focus on video content, posted on social media. According to brandwatch.com [3]:

- Facebook sees 8 billion average daily video views from 500 million users
- Snapchat users watch 6 billion videos every day
- US adults spend an average of 1 hour, 16 minutes each day watching video on digital devices
- 78% of people watch online videos every week, 55% watch every day
- 300 hours of video are uploaded to Youtube every minute
- There are 3.25 billion hours of video watched each month on Youtube
- On average, there are 1,000,000,000 mobile video views per day on Youtube
- When Instagram introduced videos, more than 5 million were shared in 24 hours

As it can be observed, video content attracts much attention and it is central to Facebook's vision for the future of the platform [4]. However, is it possible to provide an overview of all videos posted on social media at a specific time instance or interval ?

This paper proposes a novel scheme which receives at its input several different videos posted on social media and provides at its output a summary that covers the diversity of the posted content. Towards this direction, each video is analyzed and key-frames are extracted based on a correlation measure and a social computing algorithm. Then key-frames among different videos are compared so that the diversity spectrum of visual content is kept. The final summary is created by gathering the most uncorrelated frames within all extracted key-frames. Experimental results denote the full potential of the proposed scheme, its advantages as well as open issues for future work.

The rest of this paper is organized as follows: in Section 2 previous work is presented. Section 3 provides all necessary definitions. In Section 4 the fuzzy feature vector formulation scheme is described, while Section 5 focuses on key-frames extraction. Experimental results are provided in Section 6, while Section 7 concludes this paper.

## 2 Previous Work

This paper jointly examines two research areas: diversity visualization and video summarization. Regarding diversity visualization there are some representative recent works. In [5] a visual representation called the Diversity Map is proposed, which is intended to help users understand the diversity of a large set. The Diversity Map is designed to be efficiently perceived to give an accurate initial impression of a data set's overall diversity, while also allowing the user to explore relationships and interrogate the raw data using an overview as the interface. In [6] the VisBricks visualization approach is proposed, aiming at incorporating any existing visualization as a building block. This method carries attempts to break up inhomogeneous data into groups, i.e., vertically into correlated dimensions and horizontally into clusters of records, to form more homogeneous subsets, which can be visualized independently. Putting these independent visualizations of data subsets back together creates a multiform visualization, which gives an overview of the topology of the entire data set. In [7] the adaptive diversity table (ADT) is proposed to solve visual representation problems. The scheme integrates the mantra techniques to support users to accomplish seven important tasks (i.e. overview, zoom, filter, details-on-demand, relate, history, and extract) that are useful for high dimensional data exploration and data analysis. The scheme in [8] supports searching and comparing features of multivariate datasets, based on Blade Graph, which is a visualization technique for comparing distributions by emphasizing coloring according to the size of the difference. Additionally a visual analysis tool with representations is also developed for comparing data distributions.

On the other hand several methods have been proposed for video summarization. In [9] an input video is segmented into subshots using a static-transit grouping procedure. Then, entities appearing in each subshot are detected. Next the individual importance of each subshot as well as its influence on every other subshot in the original sequence are estimated. Finally, an energy function scores a candidate chain of k selected subshots according to how well it preserves both influence over time and individually important events. In [10], a "superframe" segmentation method is proposed, tailored to raw videos. Visual interestingness per superframe is estimated using a set of low-, mid- and high-level features. Based on this scoring, an optimal subset of superframes is selected to create the summary. In [11] each video is summarized by diversity ranking on the similarity graphs between images and video frames. For each video a small set of key-frames is extracted using similarity votes cast by images from the most similar photo streams. In [12] co-archetypal analysis is presented that learns canonical visual concepts by focusing on the patterns shared between video and images. Unlike archetypal analysis a regularization term is incorporated that penalizes the deviation between the factorizations of video and images with respect to the co-archetypes. In [13] an unsupervised framework is proposed that learns jointly from both visual and independently-drawn non-visual data sources for discovering meaningful latent structure of surveillance video data. A mechanism is also proposed to tolerate with missing and incomplete data from different sources. There are also several other schemes related to unsupervised video summarization, assessing the importance of frames using visual attention [14], interestingness [15], user engagement [16], content frequency [17] and non-

redundancy [18], [19]. However the diversity visualization schemes do not propose any video analysis methods, while the video summarization approaches do not focus on diversity visualization of big data. The proposed scheme tries to effectively join these two research areas and open new horizons to social media video visualization.

# 3 Definitions & Problem Formulation

Let $U = \{1, 2, \ldots, N_U\}$ be the index set of all the users of a social network of $N_U$ users, and thus $u_i$ is the $i_{th}$ user of this social network. Let $F_i$ be the index set of all of the $N_F$ friends of $u_i$ and thus $f_{ij}$ is the $j_{th}$ friend of $u_i$. Let also $I_i$ be the index set of all of the $N_I$ items posted by $u_i$ and thus $i_{ij}$ is the $j_{th}$ item posted by $u_i$.

**Definition 1**. Let $l_{i,j}$, $p_{i,j}$ and $c_{i,j}$, be respectively the corresponding likes, shares and comments item $j$, posted from user $i$, has received. If user $i$ has $N_F$ friends, then:

$$\mathbf{l}_{i,j} = [l_{i,j,f_{i1}}, l_{i,j,f_{i2}}, \ldots, l_{i,j,f_{iN_F}}, l_{i,j,f_{i(N_F+1)}}] \quad (1)$$

$$\mathbf{p}_{i,j} = [p_{i,j,f_{i1}}, p_{i,j,f_{i2}}, \ldots, p_{i,j,f_{iN_F}}, p_{i,j,f_{i(N_F+1)}}] \quad (2)$$

$$\mathbf{c}_{i,j} = [c_{i,j,f_{i1}}, c_{i,j,f_{i2}}, \ldots, c_{i,j,f_{iN_F}}, c_{i,j,f_{i(N_F+1)}}] \quad (3)$$

where $l_{i,j,f_{ik}}$ equals to 1/0 if friend $f_{ik}$ has/has not liked the respective item and similarly $p_{i,j,f_{ik}}$ equals to 1/0 if friend $f_{ik}$ has/has not shared the respective item. At the same time $c_{i,j,f_{ik}}$ equals to the number of comments friend $f_{ik}$ has made to the respective item, while $l_{i,j,f_{i(N_F+1)}}$, $p_{i,j,f_{i(N_F+1)}}$ and $c_{i,j,f_{i(N_F+1)}}$ are used to count respectively the likes, shares and comments the item $j$ has received from everybody else who is not a friend of user $i$. A slight abuse of notation is already tolerated here, an $i$ as a subscript usually refers to signify a user, where a $j$ as a subscript usually refers to signify a friend $f_{ij}$, an item $i_{ij}$ etc. relevant to this user. Finally capital $N$s are used to signify cardinalities of users ($N_U$), items ($N_I$), friends ($N_F$) etc.

**Definition 2.** Let us denote as $L_{i,j}$, $P_{i,j}$ and $C_{i,j}$ three scalar variables that count the total number of likes, shares and comments an item $j$ on $u_i$`s wall has received respectively, as the $l_1$ norms of their respective vectors (i.e. a summation of their coordinates) as:

$$L_{i,j} = \left\|\mathbf{l}_{ij}\right\|_1 = \sum_{k=1}^{N_F+1} l_{i,j,k} \quad (4)$$

$$P_{i,j} = \left\|\mathbf{p}_{ij}\right\|_1 = \sum_{k=1}^{N_F+1} p_{i,j,k} \quad (5)$$

$$C_{i,j} = \left\|\mathbf{c}_{ij}\right\|_1 = \sum_{k=1}^{N_F+1} c_{i,j,k} \quad (6)$$

Now let us assume that several videos have been posted on social media at a specific time instance or time interval. These videos have different characteristics regarding duration, illumination, motion, color, texture, theme, content, attention they have received etc. In this paper we focus on their visual diversity, aiming at providing a visual summary of this variety. By this way, a social media user will be able to see the whole spectrum of different colors, textures, motions and illuminations of posted videos. Towards this direction, in this pilot research, summarization of multiple video clips is accomplished by extracting key-frames (KFs) from each clip, representative key-frames (RKFs) from the set of key-frames and by mixing all RKFs to produce the final summary. The number of key-frames to be extracted from each clip is estimated according to the attention each clip has attracted on social media. For example, if user $u_r$ has posted a clip on his/her wall and the clip has received 160 likes, 33 comments and 8 shares, while user $u_t$ has posted another clip on his/her wall and the clip has received 455 likes, 88 comments and 26 shares, more key-frames should be extracted from the clip of $u_t$.

# 4 Fuzzy Formulation of Feature Vectors

Initially each video clip is analyzed and features are extracted for each frame using the method proposed in [20]. However all extracted features (color, texture etc) cannot be directly included in a vector, since their size differs between frames. For example, a frame consisting of twenty segments requires twice the number of feature elements than a frame consisting of ten segments. Moreover, no correspondence can be established between the elements of the feature vectors of two frames, making any comparison unfeasible. To overcome this problem, we classify color as well as texture segments into pre-determined classes, forming a multidimensional histogram. Each feature vector element corresponds to a specific feature class (equivalent to a histogram bin) and contains the number of segments that belong to this class. Segment size and location are also considered as separate feature classes. For example, a large moving segment is classified to a different feature class from a small moving segment.

In order to reduce the possibility of classifying two similar segments to different classes, causing erroneous comparisons, a degree of membership is

allocated to each class, resulting in a fuzzy classification formulation [21]. In conventional histograms, each sample – i.e., segment, in our case - may belong only to one histogram bin, so that two similar samples, located, say, in opposite sides of the boundary of two bins, are considered to belong to different bins. Using fuzzy classification, each sample is allowed to belong to several (or all) classes, but with different degrees of membership. Therefore, in the previous example, the two similar samples would slightly differ in their degrees of membership with respect to the two adjacent bins.

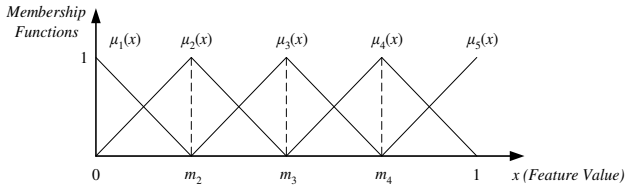


**Figure 1**: Fuzzy classification using five triangular membership functions.

Let us first consider the simple case of a one-dimensional feature $s$, e.g., the area of an image segment, taking values in a domain, which, without loss of generality, is assumed to be normalized in the interval [0,1]. This domain is partitioned, or quantized, into $Q$ classes by means of $Q$ membership functions $\mu_n(s)$, $n$=1,2,…,$Q$. For a given real value $s$, $\mu_n(s)$ denotes the degree of membership of $s$ in the $n$-th class. The membership functions $\mu_n(s)$, $n$=1,2,…,$Q$ take values in the range [0,1], so that values of $\mu_n(s)$ near unity (zero) indicate that the degree of membership of feature $s$ in the $n$-th class is high (low). The most common membership functions are the triangular ones, which involve simple calculations and they are defined as

$$\mu_n(s) = \begin{cases} 1 - 2|s - m_n|/\gamma, & |s - m_n| < \gamma/2 \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

for $n$=1,2,…,$Q$, where $\gamma$ is the width of each triangle base and $m_n$=($n$-1)/($Q$-1) is the center of each triangle, so that $m_1$=0 and $m_Q$=1. An example of fuzzy classification using $Q$=5 triangular membership functions of width $\gamma$=2/($Q$-1) is depicted in Figure 1.

It can be seen that width $\gamma$ controls the overlap between successive partitions, indicating how vague the classification is. In our case 50% overlap is used. Using this partition or quantization scheme, a fuzzy histogram can be constructed from a large number of feature samples $s_i$, $i$=1,…,$K$, each of which corresponds to an image segment, where $K$ denotes the total number of segments. Then, the value of the fuzzy histogram, say, $H(n)$ corresponding to the $n$-th class is defined as:

$$H(n) = \frac{1}{K}\sum_{i=1}^{K}\mu_n(s_i), \quad n = 1,2,\ldots,Q \quad (8)$$

We should note that the above definition reduces to the definition of conventional histograms if membership functions take binary values (0 or 1). Since, however, each sample value has non-zero degree of membership to more than one classes, the histogram can be meaningful even when the number of samples is small. Fuzzy representation thus permits the construction of histograms from a very limited set of data. This is very important since the number of segments in a frame, $K$, is typically much smaller than the total number of classes.

In the more general case of more than one segment features, including size, location, color and depth, a multidimensional feature vector is constructed for each segment. In particular, for each segment $S_i$, $i$=1,…,$K$, an $L$×1 vector $\mathbf{s}_i$ is formed as follows:

$$\mathbf{s}_i = [\mathbf{c}^T(S_i)\ d(S_i)\ \mathbf{l}^T(S_i)\ a(S_i)]^T \quad (9)$$

where the 3×1 vector $c$ includes the average values of the color components of the segment, $d$ is its texture and $a$ is its size. In a similar way, $l$ is a 2×1 vector indicating the horizontal and vertical location of the segment center. Thus, each vector has 7 elements ($L$ = 7).

According to the above, let us rewrite as $\mathbf{s}_i = [s_{i,1}\ s_{i,2}\ \ldots\ s_{i,L}]^T$, $i$= 1,2,…,$K$, the vector describing segment $S_i$, where $K$ is the total number of segments. Then, the domain of each element $s_{i,j}$, $j$=1,2,…,$L$ of vector $\mathbf{s}_i$ is partitioned into $Q$ regions by means of $Q$ membership functions $\mu_{n_j}(s_{i,j})$, $n_j$=1,2,…,$Q$. As in the one-dimensional case, for a given real value of $s_{i,j}$, $\mu_{n_j}(s_{i,j})$ denotes the degree of membership of element $s_{i,j}$ to the class with index $n_j$. Gathering class indices $n_j$ for all elements $j$=1,2,…,$L$, an $L$-dimensional class $\mathbf{n}$=$[n_1\ n_2\ \ldots\ n_L]^T$ is defined. Then, the degree of membership of each vector $\mathbf{s}_i$ to class $\mathbf{n}$ can be performed through a product of membership functions of all individual elements $s_{i,j}$ of $\mathbf{s}_i$ to the respective elements $n_j$ of $\mathbf{n}$:

$$\mu_{\mathbf{n}}(\mathbf{s}_i) = \prod_{j=1}^{L}\mu_{n_j}(s_{i,j}) \quad (10)$$

Vector $\mathbf{s}_i$ belongs to class $\mathbf{n}$, only if all its elements $s_{i,j}$ belong to the respective classes $n_j$. The membership functions $\mu_{n_j}(s_{i,j})$ should thus be combined with the "AND" operator, which is most

commonly represented by multiplication in fuzzy logic.

A simple example of 2-dimensional vectors is illustrated in Figure 2. Assume that a segment $S$ is described here by vector $\mathbf{s} = [s_1\ s_2]^T$, and $Q = 2$ membership functions $\mu_1(s_j)$ and $\mu_2(s_j)$ are used to quantize both elements $s_j$, $j = 1,2$, of $\mathbf{s}$. Since $\mu_1(s_j)$ is used to express "low" values of $s_j$ and $\mu_2(s_j)$ to express "high" values of $s_j$, we can denote classes $n_j$ as 'L' and 'H' and the two membership functions as $\mu_L(s_j)$ and $\mu_H(s_j)$. The 2-dimensional classes $\mathbf{n} = [n_1\ n_2]^T$ can then be denoted as 'LL', 'LH', 'HL' and 'HH', and the degree of membership of vector $\mathbf{s}$ to class $\mathbf{n}$ is $\mu_{\mathbf{n}}(\mathbf{s}) = \mu_{n_1}(s_1)\mu_{n_2}(s_2)$, or, taking all combinations, $\mu_{LL}(\mathbf{s}) = \mu_L(s_1)\mu_L(s_2)$, $\mu_{LH}(\mathbf{s}) = \mu_L(s_1)\mu_H(s_2)$, $\mu_{HL}(\mathbf{s}) = \mu_H(s_1)\mu_L(s_2)$ and $\mu_{HH}(\mathbf{s}) = \mu_H(s_1)\mu_H(s_2)$.

It is now possible to construct a multi-dimensional fuzzy histogram from the segment feature samples $\mathbf{s}_i$, $i = 1,\ldots,K$, exactly as in the one-dimensional case. The value of the fuzzy histogram, $H(\mathbf{n})$, is defined similarly as the sum, over all segments, of the corresponding degrees of membership:

$$H(\mathbf{n}) = \frac{1}{K}\sum_{i=1}^{K}\mu_{\mathbf{n}}(\mathbf{s}_i) = \frac{1}{K}\sum_{i=1}^{K}\prod_{j=1}^{L}\mu_{n_j}(s_{i,j}) \qquad (11)$$

$H(\mathbf{n})$ thus can be viewed as a degree of membership of a whole frame to class $\mathbf{n}$. A frame feature vector $\mathbf{f}$ is then formed by gathering values of $H(\mathbf{n})$ for all classes $\mathbf{n}$, i.e., for all combinations of indices, resulting in a total of $M=Q^L$ feature elements: $\mathbf{f} = [f_1\ f_2\ \ldots\ f_M]^T$. In particular, an index function is defined which maps the $M$ feature vector elements into an integer between 1 and $M=Q^L$,

$$z(\mathbf{n}) = 1 + \sum_{j=1}^{L} n_j Q^{L-j} \qquad (12)$$

Then, the feature vector $\mathbf{f}$ corresponding to the whole frame is a vector of length $M$, whose elements $f_i$, $i=1,\ldots,M$, are calculated as $f_{z(\mathbf{n})} = H(\mathbf{n})$ for all classes $\mathbf{n}$. It should be noted that the dimension of the feature vector $\mathbf{f}$, and consequently, the computational complexity, increases exponentially with respect to the number of partitions, $Q$. Moreover, a large number of partitions does not necessarily improve the effectiveness of the key-frame extraction algorithm. On the contrary, it results in a very large number of classes, leading to "noisy" classification. Based on several experiments, we have concluded that a reasonable choice with respect to complexity and effectiveness is $Q=3$.

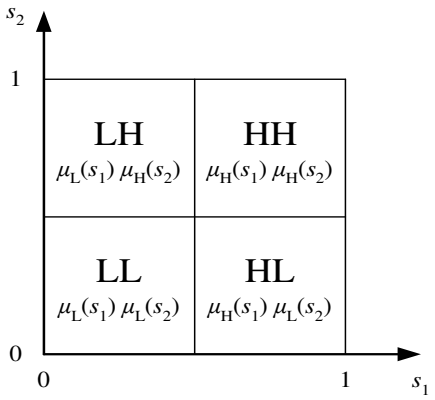


**Figure 2**: Example of two-dimensional fuzzy classification using two partitions for each dimension.

# 5 Key-Frames Extraction

In this paper, from each clip key-frames are extracted by minimizing a cross-correlation criterion, so that the selected frames are not similar to each other.

Let us denote by $\mathbf{f}_i$ the feature vector of the $i_{th}$ frame of a clip, with $i \in V=\{1, 2, \ldots, N_F\}$ where $N_F$ is the total number of frames of the given clip. Let us also denote by $K_F$ the number of key-frames that should be selected from the given clip.

The correlation coefficient of the feature vectors $\mathbf{f}_i, \mathbf{f}_j$ is defined as $\rho_{ij} = C_{ij}/(\sigma_i\sigma_j)$ where $C_{ij} = (\mathbf{f}_i - \mathbf{m})^T(\mathbf{f}_j - \mathbf{m})$ is the covariance of the two vectors, $\mathbf{m} = \sum_{i=1}^{N_F}\mathbf{f}_i / N_F$ is the average feature vector of the shot and $\sigma_i^2 = C_{ii}$ is the variance of $\mathbf{f}_i$. In order to define a measure of correlation between $K_F$ feature vectors, we first define the *index* vector $\mathbf{a} = (a_1,\ldots,a_{K_F}) \in U \subset V^{K_F}$ where:

$$U = \{(a_1,\ldots,a_{K_F}) \in V^{K_F} : a_1 < \cdots < a_{K_F}\} \qquad (13)$$

is the subset of $V^{K_F}$ which contains all sorted index vectors $\mathbf{a}$. Thus, each index vector $\mathbf{a} = (a_1,\ldots,a_{K_F})$ corresponds to a set of frame numbers. The *correlation measure* of the feature vectors $\mathbf{f}_i$, $i = a_1,\ldots,a_{K_F}$ is then defined as

$$R_F(\mathbf{a}) = R_F(a_1,\ldots,a_{K_F}) = \\ = \frac{2}{K_F(K_F-1)}\sum_{i=1}^{K_F-1}\sum_{j=i+1}^{K_F}(\rho_{a_i,a_j})^2 \qquad (14)$$

Based on the above definitions, it is clear that searching for a set of $K_F$ minimally correlated

feature vectors is equivalent to searching for an index vector $\mathbf{a}$ that minimizes $R_F(\mathbf{a})$. Searching is limited in the subset $U$, since index vectors are used in order to construct sets of feature vectors, therefore any permutations of the elements of $\mathbf{a}$ will result in the same sets. The set of the $K_F$ least correlated feature vectors, corresponding to the $K_F$ key frames, is thus represented by

$$\hat{\mathbf{a}} = (\hat{a}_1, \ldots, \hat{a}_{N_F}) = \arg\min_{\mathbf{a} \in W} R_F(\mathbf{a}) \quad (15)$$

Unfortunately, the complexity of an exhaustive search for the minimum value of $R_F(\mathbf{a})$ is such that a direct implementation would be practically unfeasible, since the multidimensional space $U$ includes all possible sets (combinations) of frames. For this reason the genetic algorithm approach of [20] is incorporated.

### 5.1 Number of Key-Frames per Clip & Extraction of Representative Key-Frames

According to the aforementioned notation, $K_F$ is the number of KFs that should be extracted for a given clip. Obviously, different numbers of KFs should be extracted for different clips. In order to estimate the number of KFs for a given clip, a social computing approach is proposed in this paper. In particular, the number of KFs for each clip depends on: (a) the social attention it has attracted (b) its duration and (c) the social attention that the other video clips of the dataset have attracted. The more attention a clip has attracted, the more KFs should be extracted, depending always on the attention that the other clips have attracted. In other words, a clip which has attracted 500 interactions when most of the other clips have attracted more interactions, should not provide many KFs. Furthermore a clip which has attracted 50 interactions when most of the other clips of the dataset have attracted fewer interactions should provide many KFs. Additionally, for the same levels of attention, a clip with significantly longer duration should provide more KFs.

For simplicity reasons let us assume that clip $i$, $i$ = 1, ….$n$, has received $L_i$ likes, $C_i$ comments and it has been shared $P_i$ times. By taking into consideration the aforementioned rules, for each clip $i$ we estimate the following parameter:

$$Q_{V_i} = \frac{L_i}{\sum_{i=1}^{n} L_i / n} + \frac{P_i}{\sum_{i=1}^{n} P_i / n} + \frac{C_i}{\sum_{i=1}^{n} C_i / n}, i = 1, \ldots, n \quad (16)$$

Here it should be mentioned that likes, shares and comments are considered of equal importance. However different settings can also be examined. After estimating $Q_V$ for each clip, clips are shorted based on their $Q_V$ value, from maximum to minimum. The video clip possessing the minimum $Q_V$ value provides only 2 KFs (since the minimum number of KFs for estimating the correlation measure is 2). Then the number of KFs is estimated by the following formula (assuming that index $i$ refers to the clips in sorted order according to $Q_V$ value):

$$K_{F_i} = \left[ 2 + a \frac{Q_{V_i}}{Q_{V_n}} + \beta \frac{D_i}{D_{avg}} \right], i = 1, \ldots, n-1 \quad (17)$$

where $[x]$ is the nearest integer function, $D_i$ is the duration of clip $i$, $D_{avg}$ is the average duration of the clips within the dataset and $\alpha$ and $\beta$ are parameters that control the number of KFs by closely following the $Q_V$ and $D$ values. By this way, more attractive clips of longer duration provide more KFs.

As a final step, all extracted KFs are gathered to form the KFs set. Then representative key-frames (RKFs) are selected from the set of KFs, again by minimizing Eq. (15). The extracted RKFs are gathered and put in order, based on the $Q_V$ value of the respected clip, so that to form the summary.

## 6 Experimental Results

Experiments have been performed on a PC with Intel Core i7-6700K @ 4.00GHz, 16 GB DDR4 RAM @ 3200 MHz, 2TB SSHD + 240 GB SSD hard drives. For evaluation purposes, on 04/09/16 we have recorded the wall information of 150 Facebook friends of the Online Computing Group (www.facebook.com/klimis.ntalianis.7). The recording has been performed using the intelligent wrapper of [22]. All other information has been discarded except of videos posted from 03/06/16 until 03/09/16. In total 390 videos have been gathered, providing on average 2.6 videos per friend or 4.33 videos per day. From the 390 videos, 57 videos have been excluded due to zero attention ($L_i$= $C_i$ = $P_i$=0) and the problems to Eq. (17) that zero attention causes. Thus 333 videos have been kept for further processing. The total duration of these 333 videos was 89,125 seconds (24 hours 45 minutes and 25 seconds), or 267.6 seconds per clip. The color encoding system was PAL at 25 frames/sec and frame size was 480×640 pixels. The total storage space was 7.82 GB, since all 333 videos were encoded using H.264.


**Figure 3**: (a) – (j) 10 out of 53 KFs extracted for the most attractive clip. (k)-(l) the 2 KFs of the least attractive clip.

Initially 2,228,125 frames were distinguished. According to the proposed scheme, for each frame a feature vector **f** should be formed by following the fuzzy formulation method described in Section 3. After making some preliminary experiments, we have observed that feature extraction and fuzzy feature vector formulation for each frame took 3.2 seconds on average. Moreover only color and texture information were considered. Motion information was excluded from current results, since other computationally intensive algorithms should also be employed. For this reason we have performed frames' sampling, by keeping only the first of every group of twenty frames. By this way 111,407 frames have remained and feature vectors were formulated after about 99 hours. This number looks large, but if we consider that it refers to a time interval of 90 days, it turns out that about 1.1 hour per day is needed for these 150 users.

After formulating a feature vector for each sampled frame, the key-frames extraction process has been triggered. The most attractive clip has received 204 likes, 43 comments and 11 shares, while the less attractive clips have received 1 like, 0 comments and 0 shares. On average each clip has received 31.07 likes, 5.13 comments and 0.17 shares. Based on these numbers the parameter $\alpha$ of Eq. (17) was set equal to 0.02. Additionally the longest clip lasted 1,303 seconds, the shortest 19 seconds and $\beta$ was set equal to 0.8. As a result $K_{F_{\max}}$ =53 for the most attractive clip, while $K_{F_{\min}}$ = 2 for the least attractive clip. In total, 5,741 KFs have been extracted, or 17.2 KFs per clip. The total time needed for extracting all KFs was about 51 minutes. For visualization purposes, Figure 3 provides 10 out of the 53 KFs for the most attractive clip and the 2 KFs of the least attractive clip.

Furthermore RKFs have been extracted from the set of the 5,741 KFs. Several experiments have been performed with different coverage percentages. Figure 4 provides results for a coverage percentage of 1% (57 RKFs), which is very small but it is selected due to space limitations. Here it should be noted that the 57 RKFs in Figure 4 are sorted from top-left to bottom-right based on the ranking of each clip. Additionally Figure 5 provides the clip-composition (number of frames per clip) of the final summary (RKFs). As it can be observed: (a) the most RKFs are provided by the top 80 clips, while only 11 frames are provided by the clips ranked from 81 to 333 (b) even low ranking clips may be represented at the final summary if their content is visually characteristic. This is a desired merit of the proposed scheme, since it focuses on diversity visualization of the posted video content.

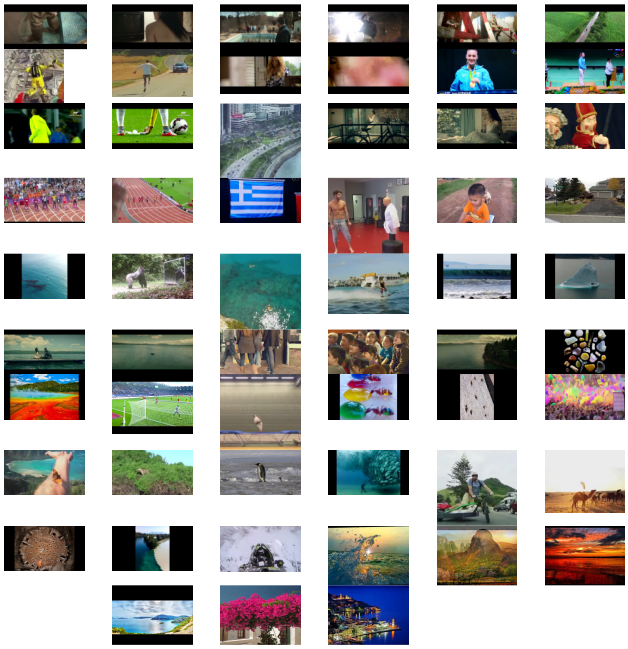

**Figure 4**: 57 RKFs (1%) sorted from top-left to bottom-right based on the ranking of each clip.

Finally, RKFs have been encoded for several different coverage percentages, using H.264. In case of 100% coverage the total size is 0.022 GB, while for 1% the total size is 0.000249 GB. Thus we achieve a reduction of the stored/transmitted information between 99.72% and 99.99% compared to the initial information.

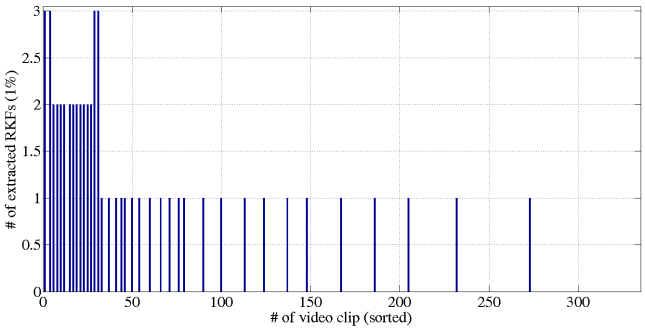


**Figure 5**: clip-composition (number of frames per clip) of the final summary (RKFs) and for 1% coverage.

## 6 Conclusion

In this paper a video content diversity visualization scheme has been proposed based on correlation. The proposed scheme assumes that several different videos are posted on social media and a novel social computing approach is described for evaluating attention. Results illustrate the promising performance of the introduced scheme.

*References:*

[1] http://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/, retrieved 08/09/2016.

[2] https://socialpilot.co/blog/125-amazing-social-media-statistics-know-2016/, retrieved 08/09/2016.

[3] https://www.brandwatch.com/2016/03/96-amazing-social-media-statistics-and-facts-for-2016/, retrieved 08/09/2016.

[4] http://www.socialmediatoday.com/marketing/top-5-facebook-video-statistics-2016-infographic, retrieved 08/09/2016.

[5] T. Pham, R. Hess, C. Ju, E. Zhang, and R. Metoyer, "Visualization of Diversity in Large Multivariate Data Sets," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 16, No. 6, November/December 2010.

[6] A. Lex, H.-J. Schulz, M. Streit, C. Partl, and D. Schmalstieg "VisBricks: Multiform Visualization of Large, Inhomogeneous Data," *IEEE Trans.Visualization and Computer Graphics*, Vol. 17, No. 12, December 2011.

[7] M.C. Wee, "An improved diversity visualization system for multivariate data," *Journal of Visualization*, Springer, doi:10.1007/s12650-016-0380-8, 2016.

[8] H. Kobayashi, H. Suzuki, and K. Misue, "A Visualization Technique to Support Searching and Comparing Features of Multivariate Datasets," *19th IEEE International Conference on Information Visualization*, July 2015.

[9] Lu and K. Grauman "Story-Driven Summarization for Egocentric Video," *CVPR* 2013.

[10] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating Summaries from User Videos," *13th European Conference on Computer Vision*, Zurich, September 2014.

[11] G. Kim, L. Sigal, and E. P. Xing, "Joint Summarization of Large-scale Collections of Web Images and Videos for Storyline Reconstruction," *CVPR* 2014.

[12] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "TVSum: Summarizing Web Videos Using Titles," *CVPR* 2015.

[13] X. Zhu, C. C. Loy, and S. Gong, "Learning from Multiple Sources for Video Summarisation," *I. J. Comp. Vis.*, Springer, Vol. 117, No. 3, p.p. 247–268, May 2016.

[14] N. Ejaz, I. Mehmood, and S. Wook Baik, "Efficient visual attention based framework for extracting key frames from videos," *Signal Proc.: Image Commun.*, Vol. 28, No. 1, 2013.

[15] M. Gygli, H. Grabner, H. Riemenschneider, F. Nater, and L. V. Gool, "The interestingness of images," In *ICCV*, 2013.

[16] W.-T. Peng, W.-T. Chu, C.-H. Chang, C.-N. Chou, W.-J. Huang, W.- Y. Chang, and Y.-P. Hung, "Editing by viewing: automatic home video summarization by viewing behavior analysis," *IEEE Multim.*, Vol. 13, No. 3, 2011.

[17] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," In *ICIP*, 1998.

[18] D. Liu, G. Hua, and T. Chen, "A hierarchical visual model for video object summarization," *PAMI*, Vol. 32, No. 12, 2010.

[19] B. Zhao and E. P. Xing, "Quasi real-time summarization for consumer videos," *CVPR*, 2014.

[20] Nikolaos D. Doulamis, Anastasios D. Doulamis, Y. Avrithis, K. Ntalianis and S. Kollias, "Efficient Summarization of Stereoscopic Video Sequences," *IEEE Trans. CSVT*, Vol. 10, No. 4, pp. 501-517, June 2000.

[21] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*, Prentice Hall, 1992.

[22] K. S. Ntalianis, N. Papadakis and P. Tomaras, "Reputation Monitoring over Rule-Stringent Social Media based on Advanced Wrapper Technologies," in *Procedia - Social and Behavioral Sciences Journal*, Elsevier, Vol. 148, p.p. 559-566, August 2014.