
# CONTENT 2017

The Ninth International Conference on Creative Content Technologies

ISBN: 978-1-61208-533-3

February 19 - 23, 2017

Athens, Greece

**CONTENT 2017 Editors**

Hans-Werner Sehring, Namics AG, Germany

# CONTENT 2017

# Forward

The Ninth International Conference on Creative Content Technologies (CONTENT 2017), held between February 19-23, 2017 in Athens, Greece, continued a series of events targeting advanced concepts, solutions and applications in producing, transmitting and managing various forms of content and their combination. Multi-cast and uni-cast content distribution, content localization, on-demand or following customer profiles are common challenges for content producers and distributors. Special processing challenges occur when dealing with social, graphic content, animation, speech, voice, image, audio, data, or image contents. Advanced producing and managing mechanisms and methodologies are now embedded in current and soon-to-be solutions.

We take here the opportunity to warmly thank all the members of the CONTENT 2017 technical program committee, as well as all the reviewers. We also kindly thank all the authors that dedicated much of their time and effort to contribute to CONTENT 2017. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

We also gratefully thank the members of the CONTENT 2017 organizing committee for their help in handling the logistics and for their work that made this professional meeting a success.

We hope that CONTENT 2017 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in the field of creative content technologies. We also hope that Athens, Greece provided a pleasant environment during the conference and everyone saved some time to enjoy the charm of the city.

**CONTENT 2017 Committee**

**CONTENT 2017 Steering Committee**
Raouf Hamzaoui, De Montfort University - Leicester, UK
Dan Tamir, Texas State University, USA
Mu-Chun Su, National Central University, Taiwan

**CONTENT 2017 Industry/Research Advisory Committee**
Hans-Werner Sehring, Namics, Germany
René Berndt, Fraunhofer Austria Research GmbH, Austria
Daniel Thalmann, Institute for Media Innovation (IMI) - Nanyang Technological University, Singapore

# CONTENT 2017

## Committee

**CONTENT Steering Committee**
Raouf Hamzaoui, De Montfort University - Leicester, UK
Dan Tamir, Texas State University, USA
Mu-Chun Su, National Central University, Taiwan

**CONTENT Industry/Research Advisory Committee**
Hans-Werner Sehring, Namics, Germany
René Berndt, Fraunhofer Austria Research GmbH, Austria
Daniel Thalmann, Institute for Media Innovation (IMI) - Nanyang Technological University, Singapore

**CONTENT 2017 Technical Program Committee**

Jose Alfredo F. Costa, Federal University - UFRN, Brazil
Leonidas Anthopoulos, University of Applied Science (TEI) of Thessaly, Greece
Kambiz Badie, Research Institute for ICT & University of Tehran, Iran
René Berndt, Fraunhofer Austria Research GmbH
Christos Bouras, University of Patras | Computer Technology Institute & Press <Diophantus> Greece
Marcelo Caetano, INESC TEC, Porto, Portugal
Juan Manuel Corchado Rodríguez, Universidad de Salamanca, Spain
João Correia, University of Coimbra, Portugal
Raffaele de Amicis, Oregon State University, USA
Rafael del Vado Vírseda, Universidad Complutense de Madrid, Spain
Myriam Desainte-Catherine, LaBRI - Université de Bordeaux, France
Klaus Drechsler, Fraunhofer-Institute for Computer Graphics Research IGD, Germany
Miao Fan, Tsinghua University, China
José Fornari, UNICAMP, Brazil
Alexander Gelbukh, Instituto Politécnico Nacional, Mexico
Afzal Godil, National Institute of Standards and Technology, USA
Seiichi Gohshi, Kogakuin University, Japan
Raouf Hamzaoui, De Montfort University, Leicester, UK
Tzung-Pei Hong, National University of Kaohsiung, Taiwan
Chih-Cheng Hung, Kennesaw State University, USA
Wilawan Inchamnan, Dhurakij Pundit University, Thailand
Pavel Izhutov, Stanford University, USA
Kimmo Kettunen, National Library of Finland | University of Helsinki
Wen-Hsing Lai, National Kaohsiung First University of Science and Technology, Taiwan

**Copyright Information**

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

**Table of Contents**

# Synesthetic Generation of Sound Clouds by Applying Social Computing

María Navarro-Cáceres*, Lucía Martín*, Javier Bajo†, Yves Demazeau‡ and Juan Manuel Corchado*
*Department of Computer Engineering and Automatics. University of Salamanca. Salamanca, Spain
Email: {maria90,luciamg,corchado}@usal.es
†Universidad Complutense of Madrid. Madrid, Spain
Email: jbajope@usal.es
‡Univ. Grenoble Alpes, CNRS, LIG
F-38000 Grenoble, France
Email: Yves.Demazeau@imag.fr

***Abstract*—The aim of social computing is to analyse the concept of social nature and design digital systems that share information between machines and users. The insights given by social computing can be applied to easily construct creative systems. As a case study, this paper presents a social machine implemented as a virtual organization where humans and machines collaborate in a creative process to transform a picture into a musical sound cloud. The prototype built from this model is evaluated by experts who rate the sounds produced following tonal music criteria.**

***Keywords–Synesthesia; Social Computing; Tonality; Music Generation; Sound Cloud***

## I. INTRODUCTION

Human beings are considered social creatures, always looking for interaction and communication with other people, and making decisions based on their social context. Social information given by such social contexts provides the basis for the inference, planning and coordination of any activity. However, this concept of a social environment cannot be translated into digital systems. In the digital world, we are socially blind [1]. Thus, the emergence of social machines has served to solve this problem and facilitate interaction and communication among people, to computerize aspects of human society, and to forecast the effects of technologies on social behavior [2].

Some authors have computed models of social intelligence based on social and psychological theories. Mission Rehearsal Exercises [3] or Tactical Language Training [4], [5] have implemented agents that develop social skills, such as leadership, foreign languages and culture in an artificial society. For example, the Sims 2 [6] is a popular game that models a virtual world with a social community. We can also consider interactive social robots, such as Teddy Bear, which was made by MIT Media Lab [7].

In the business area, the most widely used applications are recommendation systems, which suggest products, services and information to potential consumers. Companies, such as Amazon or Netflix, are adopting these systems [8] to improve customer loyalty. One approach is collaborative filtering to predict future sales by using historical sales transactions [9]. In the public sector, some government applications apply social computing to detect terrorist, criminal or other similar organizations [10], [11]. Social computing has also been applied to support decision making in health policy or state intervention [12].

With regards to music generation, some form of interaction between humans and machines is quite common. Martin *et al.* [13] presented the prototype software Toolkit to enable non-technical users to design artificial and intelligent agents to perform electronic music in collaboration with a human musician. Pachet *et al.* [14] developed The Continuator, a system able to interact with users to create a jazz improvisation in real time. Thorogood *et al.* [15] also present a system to generate soundscapes based on tweets about recent news items. There are examples of interaction with people to create different sounds or compositions through interactive evolution, such as Functional Scaffolding for Musical Composition technique [16] or neural nets [17]. Despite these proposed models of person-computer interaction, social machines have not yet been applied to this field to transform image into music.

In this work, we propose a system that supports communication among large groups of people over computer networks to generate creative content. In particular, this article focuses on uploading images by users to create a musical composition by translating colors into sound, imitating a neurological phenomenon known as synesthesia. The system is designed as a social machine described as an agent-based Virtual Organization (VO) where humans and machines collaborate in a creative process to transform a picture into a musical sound cloud. Agents start with an iterative process to extract sound from color and then generate a sound composition, denominated here as sound cloud, applying a swarm algorithm and following musical criteria such as consonance, distance between notes and distance to the main key. The prototype built from this model is evaluated by experts who rate the sound cloud or fragment produced by considering novelty and quality according to tonal music criteria.

A new architecture for creativity scenarios and an overall view of the system, based on social computing, is detailed in Section II, while the technical description of the workflow is given in Section III. Section IV presents the experiment carried out with the preliminary results obtained. Finally, Section V discusses the implications of the proposal and future work.

 ISBN: 978-1-61208-533-3 

## II. Social Machine with Virtual Organizations

Our aim is to develop a social machine capable of generating music from images provided by the users. An introductory description will be provided to give the reader a general idea about the system developed.

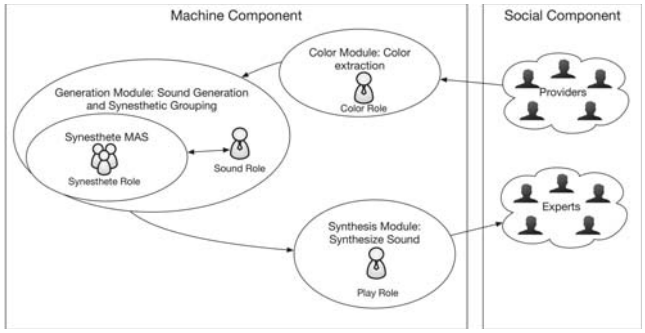



Figure 1. Social Machine schema, with human and machine components highlighted.

Fig. 1 represents a social model where human providers and experts (the social component) collaborate with the intelligent system (machine or software component). The social and software part of the system are highlighted in the squares. Each circle represents a particular stage in the workflow. Providers are focused on providing information about images, which is the input of the system. The color of the image is extracted in Hue, Saturation, Luminance (HSL) codification in the color extraction module. The HSL data of each color is then transformed into individual musical notes, and a swarm algorithm is applied to search consonant sounds, taking a sound as an individual particle with an associated color HSL, all of which occurs in the Sound Generation Module. The best sounds are selected and ordered in the Synesthetic Grouping Module following tonal music criteria. Finally, the sound is synthezised and played (Synthesizer Module) so that the Experts can evaluate the quality of the musical compositions.

The different steps described must be implemented using different intelligent modules. With an agent-based VO it is possible both to make a distributed system and easily integrate intelligent components, even combining different technologies or languages.

When developing the definition of the model used by the VO, it is necessary to analyze the needs and expectations of potential system users. The result of this analysis is the set of roles involved in the proposed model. Fig. 1 also shows the main roles identified in each previously described module; each role is represented by an agent picture.

- Provider: Represents the first part of the social machine. In this case, the user will be both the provider of the input image/picture and the listener of the final result.
- Color: Extracts colors from the image that will be associated to the Synesthete Agents.
- Synesthete: Transforms color into sound. To do so, each color is associated to a Synesthete agent. All the synesthete agents are particles immersed in a swarm algorithm that permits them to navigate through the space and change their knowledge about sounds as they are moving. At the end, different agents will be grouped according to their affinity with regards to musical aspects.
- Sound: Decides the order in which the groups of sounds corresponding to groups of synesthete agents will be played according to different parameters, such as consonance or melody leading.
- Play: Transforms the numerical notes into Musical Instrument Digital Interface (MIDI) information that can generate and play physical music.
- Experts: Given that creative process and products are hard to validate, various musical experiments have been rated following an expert evaluation. In this particular case, the output generated by the machine is evaluated by the expert interaction. The evaluation consists of rating each fragment generated on a scale from 1 (very bad) to 5 (very good). The social community can apply this form to study the quality of the compositions extracted by the system.
- Supervisor: The supervisor is a common agent in every VO. An agent who exercises this role will have overall control of the system. It analyzes the structure and syntax of all messages in and out of the system. As it is a technical agent not related with the main work, it is not represented in Fig. 1.

Section III explains the concrete implementation given for each role designed in the VO.

## III. Machine Component Description

This section details the working flow of the machine component describing the algorithms and techniques implemented. It is divided into four subsections that describe the four stages of the system. Section III-A details the color extraction of the image provided by the users and the generation of the synesthete agents, which is essential to create music. Section III-B explains the interaction model among the synesthete agents. Section III-C describes a Sound Agent that groups and orders the sound created by the synesthete agents. Finally, Section III-D presents the synthesis process to play music, carried out by the Synthesizer Agent.

### A. Color Extraction

The Color Role receives the digital image provided by the human, and then extracts the colors of the picture. In the first step, the Color agent creates a grid of cells as shown in Fig. 2.

The number of cells in the grid is set beforehand by the user, and must be an integer number lower than the number of pixels of the image. The color of each cell will correspond to the mean color between all the colors existing in the area studied. The HSL properties of the color are used to transform

 ISBN: 978-1-61208-533-3 

Figure 2. The color extraction process shown in this figure consists of three main stages.

the color into sound to instantiate the Synesthete agents (Fig. 2).

Once the color has been extracted, Synesthete agents are created and placed into a 2D space in random positions. These agents can at a future time change their positions encoded as coordinates $(x, y)$, where $x$ and $y$ are real numbers. Each Synesthete agent has the following properties:

- Color: this property consists of three attributes following the HSL model.
- Position: Considered a bidimensional position $(x, y)$ as previously explained.
- Sound: Consists of note names in terms of loudness and pitch, as we will explain below.
- Associated Sound: Sound vector of the nearest agents.
- Velocity: An array with two vectors: One for the velocity in the X axis and another for the velocity of the Y axis.
- Sounding: A Boolean variable to store the decision about whether the sound is good enough to be played.

The color of each cell (in HSL model), produced by the Color agent, is transformed into a sound by these Synesthete agents. Both pitch and loudness can be linked with certain color properties. This relation can be established in different ways. A social interaction could be used in this stage to select the color for each note; however, for the scope of this study, a standard relation proposed by Sanz [18] was followed. First, Hue is automatically associated to Note name following the Lagresille system [18], where each set of color tones corresponds to a specific note. Then, Saturation is related to Loudness. We can consider this association to be logical, thus the more intense we see the color, the more intense the sound should be. The Saturation is translated into values of Loudness from 1 to 5, where 1 is very low and corresponds to a 0 of saturation and 5 is very high volume and corresponds to a saturation of 100%. Finally, Luminance refers to the Octave. As the luminance value is increased, the sound has a higher pitch, and will therefore be more acute. In order to preserve a balance between the coherence of different notes but also diversity in octaves, this is mapped from octave number 2 to octave number 6 according to the MIDI codification.

For instance, the first cell of Fig. 2 corresponds to a HSL codification of (242, 61, 52). That is translated into G note according to Lagresille System. The loudness corresponds to the value 3.44. The Luminance corresponds to the third octave, according to the integer mapping carried out.

### B. Notes Grouping

The behavior of the synesthete agents that implement the Synesthete Role is based on a particle swarm optimization (PSO) algorithm [19]. Thus, the movement is regulated by attraction forces capable of modifying their position and velocity following a fitness function. The swarm allows the association of several agents with similar features following a fitness function, which will now be briefly described. The steps followed in this algorithm are:

1) Each agent has a position $P$ in the 2D space, and can produce one sound from the color associated.
2) Each agent $a_1$ searches its neighbor agents $a_2$, $a_3$,..., $a_N$ in the space based on Euclidean distance, and exchanges information with them to measure the quality between these sounds. This process, explained below, provokes an attraction force between the agents. The strength depends on the level of quality of the sounds.
3) These steps are repeated until a sound balance is found, upon algorithm convergence. Sound balance means that the particles do not update their positions significantly over the iterations. This indicates that the sounds are balanced in their right positions according to the quality function analyzed here.

In the final state, an agent organization with groups of pleasant sounds will be obtained.

As mentioned above, each agent rates its quality according to a fitness function. This function considers two musical factors to evaluate the quality of the sounds: consonance properties following the tonal standards and loudness, according to (1).

$$F(a, n) = \sum_{i=0}^{M} (C(x, n_i) + L(x, n_i)) \quad (1)$$

where $L(a, n_i)$ considers the loudness of the sound corresponding to agent $a$ and compares it with the loudness of $n_i$ sound, and $C(x, n_i)$ measures the quality of the intervals between the sound of $a$ and the sound of $n_i$ ($i$-neighbor). $C$ comprises a combination of consonance, distance to the main key (which is selected according to the most common note in the space) and distance to the $n_i$ musically speaking, all based in the Fourier transform (FFT) of each sound. Due to its complexity, $C$ function is not fully described here, but analysed in our previous article [20] and based on the Tonal Interval Space proposed by Bernardes *et al.* [21], which allows us to create tonal music. TIS is defined as a 12-D space, where geometrical distances captures musical properties. To do so, the FFT is extracted from each note initially codified as a chroma vector. The set of the first six components of the FFT vector, considering real and imaginary part, comprises the Tonal Interval Vector (TIV), which are the coordinates of the 12-D space. That permits to encode not only notes, but also chords or keys. Bernardes *et al.* [21] and Navarro *et al.* [20] demonstrate that Euclidean distances and other geometrical measures taken in such space, captures some musical properties. In particular, the following measures were considered here:

 ISBN: 978-1-61208-533-3 

- Consonance between two notes $n_1$ and $n_2$: In the TIS, this value is measured as the distance between the corresponding TIVs of $n_1$ and $n_2$.
- Belonging to the main key: In the TIS, we measured the degree of membership of one note to the key by calculating the angle between the projection of TIS corresponding to the key codification and the projection of TIS corresponding to the note $n_i$.
- Voice-leading: This part allows us to analyse voice leading between two notes considering not only the consonance, but also the number of semitones between them.

$C$ will be a linear combination between the first measure (consonance between two notes) and the voice-leading.

The VO behavior is inspired from Particle Swarm Optimization (PSO) behavior. This algorithm proposed by Kennedy [19] is an example of swarm intelligence where each individual is moving freely through space considering three factors: the inertia weight component, the cognitive component, and the social component. To begin, (i) inertia force is related to the physical inertia and depends on the previous force applied to the particle; (ii) cognitive components refer to the attraction forces between particles or groups of particles and, finally, (iii) the social component is related to the exchange of information among particles. Within the algorithm, the particles have several premises to accomplish within system $S$:

- Stay near the neighbor particles. This rule prevents particles from straying too far from the center of the system.
- Move towards the gravity center. Each particle is attracted by other particles depending on certain parameters previously established. Thus, attraction forces are fundamental in this type of model.
- Avoid collisions between particles. In this case, repulsive forces are needed if the distance between two particles is too small.

The next position $p_t$ of a particle $a$ in the swarm depends on the current position $p_{t-1}$, the current velocity $v_{t-1}$, the best position at the current time $pb_i$, and the best position found by any of its neighbors $pb_n$, following (2) [19]:

$$\vec{p_t} = f(\vec{p_{t-1}}, \vec{v_{t-1}}, \vec{pb_i}, \vec{pb_n}) \tag{2}$$

PSO needs to be adapted to solve our specific problem. The particles in the algorithm are represented by the Synesthete agents in the VO. The three factors that provoke the particle movement are adapted to our creative system. Thus, attraction forces are related to inertia and cognitive components, while the exchange of information between agents is the social component. The cooperative attitude in a VO is also essential to achieve the goal of the whole system.

In this case, the communication allows the agent to know about its neighbors colors, sounds and position. These agents can have cooperative and non-cooperative behavior. The cooperative interactions are based on an attraction function. The intensity of the attraction forces depends on the fitness function $F(x, y)$, explained in (1). This function permits the modification of the position of each particle according to the quality measures. In contrast, non-cooperative interactions refer to a repulsion function. This repulsion function is activated only if the agents' positions are very near each other, in order to avoid collisions following the theory presented in Blackwell *et al.* [22].

The algorithm starts when each agent searches its neighbors. To do so, a ratio is established so that the agent selects who its neighbors can be. The agents carry out an interaction process to exchange information, and finally decide the best position according to the attraction force generated. The force for Agent $a_i$ depends on the values obtained by applying the fitness function according to its neighborhood.

Equation (3) represents the calculations to get the next position $p_{t+1}$ of a given agent $a_i$ at iteration $t$. This is a linear combination of the current velocity $v_{t+1}$ and the previous position $p_t$.

$$\vec{p_{t+1}} = \vec{p_t} + \vec{v_{t+1}} \tag{3}$$

Note that all of the values referring to position and velocity are vectors. The particles $a_i$ velocity are given by (4).

$$\vec{v_{t+1}} = \sum_{k=1}^{N} F * \vec{v_t} + (\vec{p_{kt}} - \vec{p_t}) \tag{4}$$

where $k$ represents the $k$ neighbor agent present in particle $a_i$. The three different components described previously (inertia, cognitive and social) are each represented by one of the three terms in (4). The fitness function $F$ regulates the effect of the momentum (velocity) component. The vector $(p_{kt} - p_t))$ allows the movement of particles towards the best position found by all the neighbor agents.

Within each iteration, every particle moves in a direction that is determined by the influence that its neighbors have over it. In our case, unlike the general PSO algorithm, there is no global best position for the whole system in the intermediate steps. The particles move around the search space based on these equations for a number of iterations until, if all goes well, they all converge. The convergence criterion is achieved when the positions of particles are not noticeably modified. The global best can then be taken as the final solution produced by the algorithm. At the final point of the algorithm, we also expect diverse subgroups to be generated by the attraction forces.

In order to play the full composition, each agent has the ability to decide whether to sound; in other words, to modify the property of "sounding", which is a Boolean value according to musical quality factors. To achieve this, a threshold is established so that if the values for the fitness function are not above this threshold value, they are not candidates to create the melody by the Sound Role (described later), and consequently, the "sounding" property is set to 0.

 ISBN: 978-1-61208-533-3 

### C. Sound Cloud Generation

The sound cloud is generated by the Sound Agent when the Synesthete agents have been grouped. The Sound agent decides the playing order for each group. This order depends on sound expectedness, the position of the synesthete agents, and the group they belong to.

The Sound agent extracts the positions of those agents whose attribute "sounding" is set to 1, meaning that the sound can be played. It studies the subgroups that exist by applying a clustering algorithm, and according to the mean position of each cluster, it decides the sub-swarm each agent belongs to.

The first group $Sg_1$ and the first note $n_1$ is randomly chosen. The agent then applies an expectedness measure to evaluate the probability of each sound being played in a composition following the selected value $n_1$. We study this probability by using the difference in loudness between the notes and rules of classical music based on the Tonal Interval Space [21], such as voice-leading and the belonging to the key above mentioned.

To select the first note of the following sub-group $Sg_2$ chosen randomly, we have to evaluate each note in the $Sg_2$ compared with the last note chosen in $Sg_1$. Again, the note or group of notes with the best values will be the next chord in the progression. From here, the process will be repeated until all the sounds are selected from the swarm system.

The rhythm is out of the scope for this first experiment, thus the sound will be constant along the melody.

### D. Synthesizer Role

As we continue to advance in this section, the last step in our VO aims to synthesize the results proposed by the previous algorithm. The numbers for the pitches and the loudness obtained need to be interpreted so that an instrument or a synthesizer can play them. The MIDI format transforms agents properties into MIDI data [23]. This is the main task of the Synthesizer Role. Once this task is accomplished, the Role agent extracts the MIDI info and transforms it into audio information so that the computer can reproduce it.

## IV. Results and Discussion

The evaluation presented in this paper aims to investigate whether sounds with low fitness values are judged more consonant than sounds with higher penalty values; in other words, if the fitness function measures the social acceptance of the music generated.

We made a preliminary experiment deploying a social network in a specific web for a number of people. There, the members can login to upload any image and listen to the results. Curiously, almost all the images used in the social network displayed for a number of people, were personal images, reflecting some events in their personal lives, such as travels, monuments or family photos. We finally selected three picture that we considered as anonymous enough, as shown in Fig. 3.


Figure 3. Collection of pictures applied to the system to extract sound cloud music.

For the purposes of this work, we considered 43 musical experts who evaluated the individual melodies in terms of tonal musical quality and their adaptation to the image they derive from. The evaluation consists of rating each fragment presented on a scale from 1 (very bad) to 5 (very good). Fig. 4 shows the results obtained in the evaluation. The fragments can be listened in the following url: goo.gl/GpLgHw. With each fragment, the image was shown in order to validate both parameters at the same time.

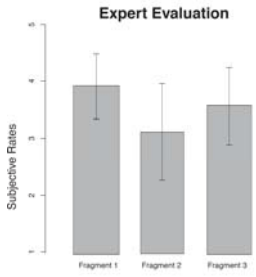



Figure 4. Evaluation results for each fragment.

In the plot, the mean punctuation of these 43 experts is shown for each image. In this case, we expected each musical fragment to be valued with a scale from 1 (very bad) to 5 (very good). Among the compositions proposed, one was well evaluated and the rest were evaluated as a fair to good sound cloud. Fragment 1 corresponds to Fig. 3a, Fragment 2 to Fig. 3b and Fragment 3 to Fig. 3c. Fragment 1 has a mean of 3.92, considered as almost Good composition according to quality and adaptation to the image. The error is about 0.51, meaning the values have been oscillating between 3.41 and 4.43, both considered above Fair rates.

Fragment 2 obtains a mean evaluation of 3.11, considered as Fair composition according to quality and adaptation to the image. However, the mean error is 0.96, meaning the values have been oscillating between 2.15, considered as bad rate and 4.43,considered as good rates. This oscilation might be due to personal preferences for the adaptation of the melody to the image, or the worse quality in musical terms comparing to the Fragment 1.

Fragment 3 gets a mean rate of 3.75 considered as almost Good composition according to quality and adaptation to the image. The error is about 0.72, meaning the values have been

 ISBN: 978-1-61208-533-3 

TABLE I. COMPARISON BETWEEN OUR PROPOSAL, JANUS SYSTEM AND KIRKE & MIRANDA'S WORK

| | Our Work | JANUS | Kirke's |
|---|---|---|---|
| Creative Products | X | - | X |
| Interaction | X | X | X |
| Open System | X | X | - |
| Growth Capacity | X | X | - |
| Social Character | X | - | - |
| Public Participation | X | - | - |
| Uses a MAS | X | X | X |
| BDI Architecture | | X | X |
| Swarm Computing | X | - | - |
| Combines reasoning with swarm | X | - | - |
| VO support | X | X | - |
| Compatible with web services | X | X | - |
| Executable in different SO | X | X | X |
| Support to experts to interact with the system | X | X | X |
| Charge Balance | X | X | - |
| Provides a user interface | X | - | X |
| Provides a logging tool | - | X | - |

oscillating between 3.03 and 4.47, both considered above Fair rates.

It is necessary to consider that the sound cloud is not expected to be a full tonal piece, although it follows the main tonal standards, but a soundscape or an ambient piece inspired by a painting. Additionally, all the rates obtained are subjective evaluations, which depends on the social culture, mood and personal preferences and perceptions. Therefore, the results for the same fragment can be very different between individuals. However, in view of the present results, our social machine is able to provide acceptable compositions that in some way reflect the colors of a pictorial work.

We also present a comparison among three systems to highlight the advantages of our work. In particular, Kirke's work, a creative work based on MAS [24] along with the JANUS System [25], a framework that works with VOs and MAS for general purposes. The qualitative comparison is shown in Table I. It is worth noting that the study developed by Kirke *et al.* [24] uses emotions and MAS to generate a musical melody. The use of emotions to create new music enhances the systems originality; however, while it interacts with the users, it is not a proper application for social communities. Moreover, scalability and flexibility are not included in their work, as they did not design the system following a VO methodology.

JANUS [25] is a multiagent platform that was specifically designed to deal with the implementation and deployment of multiagent systems. It is based on an organizational approach and is focused on supporting the implementation of the concepts of role and organizations as first-class entities. This consideration has a significant impact on agent implementation and allows an agent to easily and dynamically change its behavior. This feature is also shared with our system, supporting the VO design along with the musical composition. Although BDI architecture is not considered in this specific work because the agents designed do not require such architecture beforehand, it is a key feature to consider in a future work, to make a general framework that supports creative process. Apart from this, our system shares the majority of the features corresponding to a VO framework, adding a social component essential for the success of a composition system.

## V. CONCLUSIONS

A social machine was developed to transform colors into music. This model was implemented with a VO to create a flexible system that can be adapted to new social contexts given by different social situations. The social component is divided into two parts: the social providers that give the pictures to the system to extract the color, and the social experts, who evaluate the quality of the music generated.

The machine component contains a workflow of four stages, all of which are based on VOs and implemented by a multiagent system. The first step consists of the color extraction of the image provided by the social community. The color properties are used to give the initial parameters that the synesthete agents use to rate the quality of the sound generated in order to move throughout the space. The movement originated in the space follows diverse rules according to a modified swarm algorithm designed for this particular work.

In order to generate a consistent music composition, an agent is developed to evaluate the probability of each sound based on the previous sounds existing in the composition. This final output is synthesized and played for an expert community. Rhythm component is primitively developed, so we will consider improving this aspect in future work.

The opinions of these experts were used to evaluate the acceptance of the music created and their accordance to the images given as an input. In particular, we considered three images with their corresponding musical fragments that were rated by 43 experts. The community of experts agrees that the quality is acceptable for this approach of our model in view of the mean results obtained, all of them above Fair good compositions. However, the number of images and fragments is not enough to extract strong conclusions. Therefore, we plan to extend this test in a future work, adding a larger number of images and analysing for the one hand the music quality and on the other hand, the adaptation of the music to the image.

However, this opinion does not affect the machine result, as the interaction is limited to the user choosing an input (picture) and then evaluating the musical result. Therefore, the machine does not learn, removing a quite important part of the social environment, such as experts' opinions. Thus, we propose futuare work to add a feedback option to the system in order to automatically incorporate expert evaluations to improve our system. This loop will permit to be plunged into an interactive evolution, in which the machine will store each experience (image, melody and overall rates) and will consider it to train a model and extract new melodies from new cases, adapted to the social evaluations of previous experiences.

## ACKNOWLEDGMENTS

This work has been partially supported by the Spanish Government through FPU program FPU2013/2071.

 ISBN: 978-1-61208-533-3 

## REFERENCES


[1] T. Erickson and W. A. Kellogg, "Social translucence: an approach to designing systems that support social processes," ACM transactions on computer-human interaction (TOCHI), vol. 7, no. 1, 2000, pp. 59–83.

[2] F.-Y. Wang, "Toward a paradigm shift in social computing: the acp approach," Intelligent Systems, IEEE, vol. 22, no. 5, 2007, pp. 65–67.

[3] W. R. Swartout et al., "Toward virtual humans," AI Magazine, vol. 27, no. 2, 2006, p. 96.

[4] C. Girard, J. Ecalle, and A. Magnan, "Serious games as new educational tools: how effective are they? a meta-analysis of recent studies," Journal of Computer Assisted Learning, vol. 29, no. 3, 2013, pp. 207–219.

[5] M. Si, S. C. Marsella, and D. V. Pynadath, "Thespian: Modeling socially normative behavior in a decision-theoretic framework," in Intelligent Virtual Agents. Springer, 2006, pp. 369–382.

[6] P. Zaphiris and A. A. Ozok, "Human factors in online communities and social computing," Handbook of Human Factors and Ergonomics, Fourth Edition, 2012, pp. 1237–1249.

[7] W. D. Stiehl et al., "Design of a therapeutic robotic companion for relational, affective touch," in Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on. IEEE, 2005, pp. 408–415.

[8] F.-Y. Wang, K. M. Carley, D. Zeng, and W. Mao, "Social computing: From social informatics to social intelligence," Intelligent Systems, IEEE, vol. 22, no. 2, 2007, pp. 79–83.

[9] Z. Huang, D. D. Zeng, and H. Chen, "Analyzing consumer-product graphs: Empirical findings and applications in recommender systems," Management science, vol. 53, no. 7, 2007, pp. 1146–1164.

[10] E. Ferrara, P. De Meo, S. Catanese, and G. Fiumara, "Detecting criminal organizations in mobile phone networks," Expert Systems with Applications, vol. 41, no. 13, 2014, pp. 5733–5750.

[11] D. López Sánchez, J. Revuelta, and F. De la Prieta, "Twitter user clustering based on their political preferences and the louvain algorithm," in Proceedings on Practical Applications of Agents and Multi-Agent Systems. In Press. Springer Verlag, 2016.

[12] J. Bajo, J. F. De Paz, G. Villarrubia, and J. M. Corchado, "Self-organizing architecture for information fusion in distributed sensor networks," International Journal of Distributed Sensor Networks, vol. 2015, 2015, pp. 2–10.

[13] A. Martin, C. T. Jin, and O. Bown, "A toolkit for designing interactive musical agents," in Proceedings of the 23rd Australian Computer-Human Interaction Conference. ACM, 2011, pp. 194–197.

[14] F. Pachet, "The continuator: Musical interaction with style," Journal of New Music Research, vol. 32, no. 3, 2003, pp. 333–341.

[15] M. Thorogood, P. Pasquier, and A. Eigenfeldt, "Audio metaphor: Audio information retrieval for soundscape composition," Proc. of the Sound and Music Computing Cong.(SMC), 2012, pp. 277–283.

[16] A. K. Hoover, P. A. Szerlip, and K. O. Stanley, http://maestrogenesis.org/, 2012, online; accessed December 10, 2016.

[17] J. Ye and S. Chen, https://www.cs.swarthmore.edu/~meeden/cs81/s14/papers/AndyLucas.pdf, 2014, online; accessed December 10, 2016.

[18] J. C. Sanz, Lenguaje Del Color: Sinestesia cromática en poesía y arte visual. H. Blume, 2009.

[19] J. Kennedy, "Particle swarm optimization," in Encyclopedia of Machine Learning. Springer, 2010, pp. 760–766.

[20] M. Navarro-Caceres, M. Caetano, G. Bernardes, and J. M. Corchado, "Iterative generation of chord progressions in the tonal interval space with an artificial immune system," Expert Systems with Applications, 2016, p. In review.

[21] G. Bernardes, D. Cocharro, M. Caetano, C. Guedes, and M. Davies, "A multi-level tonal interval space for modelling pitch relatedness and musical consonance," Journal of New Music Research, 2016.

[22] T. Blackwell, "Swarm music: improvised music with multi-swarms," Artificial Intelligence and the Simulation of Behaviour, University of Wales, 2003.

[23] W. B. Hewlett and E. Selfridge-Field, "Midi," in Beyond MIDI. MIT Press, 1997, pp. 41–70.

[24] A. Kirke and E. Miranda, "A multi-agent emotional society whose melodies represent its emergent social hierarchy and are generated by agent communications," Journal of Artificial Societies and Social Simulation, vol. 18, no. 2, 2015, p. 16.

[25] N. Gaud, S. Galland, V. Hilaire, and A. Koukam, "An organizational platform for holonic and multiagent systems," in Proceedings of Sixth international Workshop on Programming Multi-Agent Systems, 2008, pp. 104–119.


 ISBN: 978-1-61208-533-3 

# Localized Content Management with the Minimalistic Meta Modeling Language

Hans-Werner Sehring
Namics
Hamburg, Germany
e-mail: hans-werner.sehring@namics.com

*Abstract*—**Content management systems are in widespread use for document production. In particular, we see the pervasive application of web content management systems for web sites. These systems serve authors that produce content and web site users that perceive content in the form of documents. Today, one focus lies on the consideration of the context of the web site user. Context is considered in order to serve users' information needs best. Many applications, e.g., marketing sites, focus on making the user experience most enjoyable. To this end, content is directed at the users' environmental and cultural background. This includes, first and foremost, the native language of the user. Practically, all respective web sites are offered in multiple languages and, therefore, multilingual content management is very common today. Content and its structure need to be prepared by authors for the different contexts, languages in this case. Contemporary content management system products, though, each follow different approaches to model context. There is no single agreed-upon approach because the different ways of multilingual content management have different focuses. This paper discusses different aspects of multilingual content management and publication. The Minimalistic Meta Modeling Language is well suited for context-aware content management. This paper demonstrates how this modeling language can be used to support a universal approach to multilingual content management. This approach allows content modeling without consideration of product properties. This way, it takes away constraints from content modeling and it removes dependencies to content management system products.**

***Keywords-content management; web site management; multilingual content management; multilinguality; context-awareness.***

## I. INTRODUCTION

Web sites are operated by nearly all organizations and enterprises, and they are created for various purposes. The management of such sites has evolved to *content management* that separates web site content, structure, and layout. This way, content can be published on different media and on different channels. Certain parameters can influence the production of documents from content, e.g., the viewing device used by the user or her or his current context.

Consequently, most web sites are produced by a *content management system* (CMS), a web CMS in this case. Currently, web sites increasingly exhibit consideration of content that is tailored to the context of the content's percipient. They do so either to provide content with maximal value to the visitor, in order to convey a message best, to present a company in the best possible way, etc.

The most basic contextual property, to this end, is the native language of the consumer. Content should be presented to the user in this language. At least, textual content is translated. More advanced approaches take the culture and the habits of a user into account.

(Written) Language has an impact on layout. E.g., there need to be web page layouts for languages written from left to right and for ones written from the right to the left. On top of that, the writing direction has an impact on, e.g., the placement of navigation and search elements on a web page.

Some time ago, multilingual web appearances and print publications were identified as a major challenge for organizations [1]. In practice today, the problem is addressed by various approaches in different CMS products. However, there is no systematic consideration of these approaches and the characteristics of the resulting solutions. This leads to the content management approach taken to be dependent on the CMS product chosen for a particular web site appearance.

The rest of this paper is organized as follows. Section II defines requirements for multilingual web sites and the CMSs producing them. Section III describes related approaches to multilingual web site production. Section IV briefly introduces the Minimalistic Meta Modeling Language. Section V discusses the application of this language for multilingual content management. The conclusions and acknowledgement close the paper.

## II. LOCALIZED CONTENT MANAGEMENT REQUIREMENTS

The general approach to multilingual web site production is similar for most approaches.

Its foundation is language- and country-independent content, or at least content storage organized in a way that allows content to be localized easily. To this end, an initial *internationalization* (often abbreviated as I18n) step removes all cultural assumptions from content.

On that basis, a *localization* (L10n) procedure adapts content for a specific country, region, or language.

There are many considerations that have to be taken into account to enable this basic process. On top of the linguistic and cultural tasks of localization, there are business considerations and technical issues about the management of content, its structure, and organization (its physical structure) [2]. Content management processes for localization have to be defined. In this paper, we concentrate on the technical issue of multilingual content representation.

### A. Basic Multilingual Content Management Strategies

There are three typical strategies for the management of multilingual and multicultural content [3]:

 ISBN: 978-1-61208-533-3 

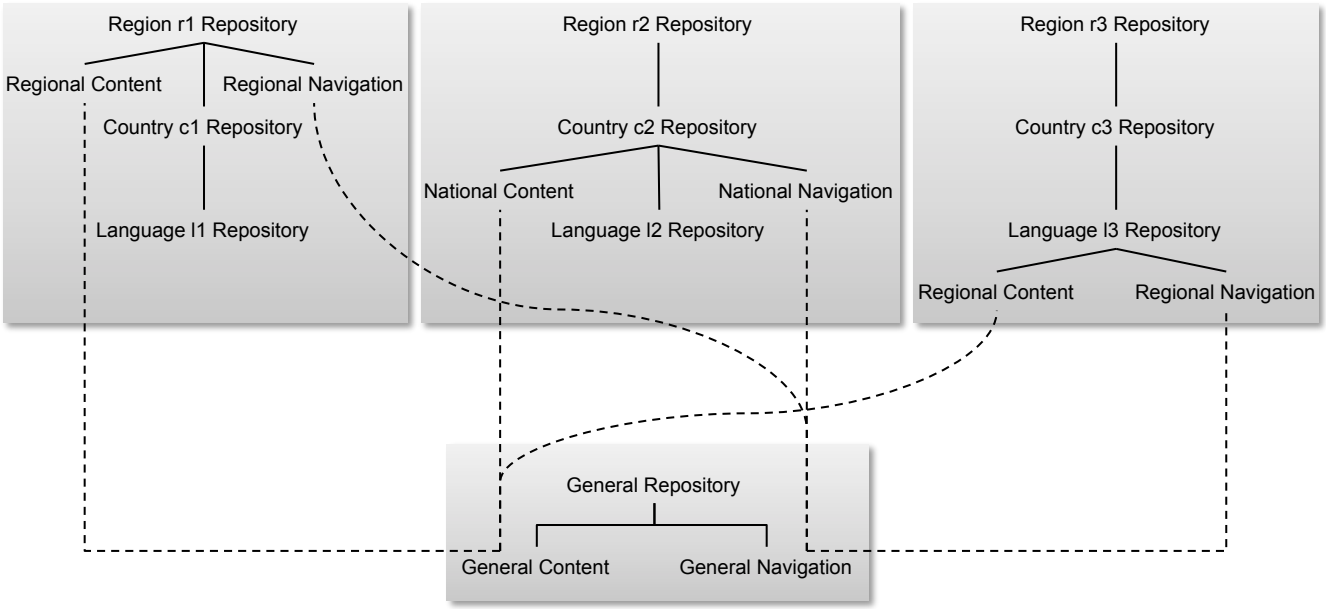



Figure 1. Example of a content repository organization for multilingual content with content distribution on different levels.

*1) Central control over the content:* Content is distributed by a central authority and it is translated to different target languages, but it is typically not adapted in other ways. I.e., there are no structural changes, and the layouts are not adapted to local preferences.

*2) Decentralized management of multiple local sites without coordination:* The local sites typically use a localized design. This approach does not ensure homogeneous quality in all localized appearances, there is no means to enforce content to be current in all local repositories, and there is no way to grant a globally recognizable web site standard, e.g., a corporate design.

*3) A hybrid approach of the first two:* It allows dealing with global, regional, and local content. Global content is produced centrally and translated for global use. Regional content is localized from centrally provided content, but is also adapted to and used in a regional context. Local content is produced locally in the local language in addition to global and regional content.

Because of the possibly combined advantages, many organizations favor the third approach. It requires tool support that is discussed in the subsequent subsections.

In practice, there are basically two CMS setups that correspond to centralized and decentralized content management: a central multi-tenant CMS that allows hosting multiple sites and relating these to each other, and isolated local CMSs that exchange content while providing their own web site structure and layout. The subsequent subsections of this section discuss these two approaches.

Fig. 1 shows an overview over different exemplary content repositories organized according to the two ways of multisite content management. Each repository represents one CMS instance or one content collection inside a CMS together with its structure, layouts, etc.

The three repositories at the top of Fig. 1 show content localization at different levels in content trees of multisite CMSs. In this example, each CMS hosts collections for regional, national, and language-specific content. These are just three arbitrary levels of content collections. The solid lines in the figure denote relationships between collections where the lower one is derived from the upper one.

The *General Repository* at the bottom represents the pool of internationalized content that is used for content distribution from a central content pool. The dashed lines represent content passed from one repository to another.

The sample repositories in Fig. 1 contain content (text, images, etc.), as well as navigation nodes and structure. These parts of the repositories are only shown where needed.

Maintaining content consistency across different localized versions is time consuming and error prone. There are two primary ways of content distribution and localization: manual and semiautomatic [2]. These apply to both approaches, centralized as well as decentralized CMSs.

### B. Related Content

Professional CMSs allow defining content collections and relating them to each other. A typical pattern is a master-variant model, where a *variant* can be derived from every piece of content. The original content then plays the role of a *master*. Whenever the master content changes, some actions on the variants are induced.

When the master is extended with additional content, e.g., new substructures, then it may provide *default* or *fallback* values to the variants. Often the English version of a web site is chosen as a master, so that new content that is not yet localized shows up in English on the various sites.

 ISBN: 978-1-61208-533-3 

Fallbacks are problematic for composite documents, e.g., images embedded in a text [4]. When an image is updated, this may, e.g., result in an English image contained in a French text. An application-specific fallback logic may be needed for composites.

Additionally, changes to the master may result in translation workflows being started. Such a workflow either demands that new or changed content is translated manually, or it employs automatic translation tools to create localized content.

Manual translation typically has to be performed by professional translators. To enable these to work with a CMS, there are content interchange formats like the XML Localisation Interchange File Format (XLIFF) and Translation Memory eXchange (TMX) for output and input of multilingual content.

During automatic localization there are easy translation tasks like adaptations of number formats, measurement units, currencies, etc. From a cultural viewpoint there is no general answer to the question whether a document's content can be changed while retaining its structure, though [5]. In general, only the translation of content according to a centrally given structure is achievable.

Scientific approaches to automatic translation are based on semantic models of content, often ontology-based [6].

In any case, it is crucial for editors to learn how to prepare content in a way that is suitable for localization [7].

### C. Independent Content

In a decentralized approach, CMSs maintain local content and structures. Localization may be performed by translation of internationalized content that is provided in a central content pool, by adding new local content, and by omitting centrally provided content from a local repository.

This scenario furthermore gives single CMSs complete freedom concerning the visualization of content.

Since a central repository provides base content, the decentralized approach requires means to ship content from that central instance to the local repositories. For content collections inside one CMS, the shipping might just consist of internal references. If separate CMS instances need to exchange content, some external content format is required.

As indicated in Fig. 1, the repositories might form a hierarchical network of content pools, ranging from global over regional down to local repositories. Though these hierarchies result from the master-variant relationships (see previous subsection), content shipping should take hierarchy levels into consideration (see the dashed lines in Fig. 1).

## III. Related Work

We briefly discuss related approaches to multilingual content management and commercial CMS products.

### A. Modeling Approaches

Typically, the management of multilingual web sites relies on a CMS. There are approaches to solve multilingual content management on the level of HTML files, though.

MultiLingual XHTML (MLHTML) [8] is an extension to HTML. It was designed to include content for different languages in the same page file. An XSL style sheet is used to transform it to a plain HTML page for a given language. The approach is well suited for static sites without a CMS in the background and for large sets of existing static HTML pages. It requires web sites to have the same structure and the same layout across all languages, though.

### B. Content Management System Products

Professional CMS products support multilingual content management. To name some examples, Adobe Experience Manager (AEM), CoreMedia CMS, and Sitecore all follow a master-variant approach to multisite management. They provide functionality to create a deep copy of a master site. The content entities from the copy are automatically related to the corresponding master entities.

All products allow local editing of content copies, and changes to the master lead to notifications sent to editors. CoreMedia also allows editing the content's structure. Sitecore manages navigation structures locally.

Some products add workflow tasks for the translation of all content entities. Workflows may drive automatic or manual translation processes. Some of the products provide workflows for external translations using XLIFF.

The master site serves as a fallback for missing localized content. To this end, AEM and CoreMedia allow to freely choose the master. Sitecore prefers US English for master content. Instead, Sitecore provides fallback chains to, e.g., have a series of fallback languages before using the master.

## IV. M3L

The *Minimalistic Meta Modeling Language* (*M3L*, pronounced "mel") is a modeling language that is applicable to a range of modeling tasks. It proved particularly useful for context-aware content modeling [9].

In order to be able to discuss multilingual content management with M3L in the subsequent section, we briefly introduce the M3L modeling constructs.

M3L offers a rather minimalistic syntax that is described by the following slightly simplified grammar (in EBNF):

*model* ::= ⟨*def-list*⟩
*def* ::= ⟨*ref*⟩ "*is*" ⟨*id-list*⟩
["{"⟨*def-list*⟩"}" [⟨*production-rule*⟩]
| ⟨*production-rule*⟩ | ";" ]
*ref* ::= ⟨*id*⟩ ["from" ⟨*ref*⟩]
*id-list* ::= ("a" | "an" | "the") ⟨*ref*⟩ ["," ⟨*id-list*⟩]
*def-list* ::= ⟨*def*⟩ [⟨*def-list*⟩]
*production-rule* ::= ("|=" ⟨*def*⟩ | "|-"{⟨*ref*⟩|*string*}) ";"

The production for identifiers (*id*) is omitted here. It is a typical lexical rule that defines identifiers as character sequences. Identifiers may—in contrast to typical formal languages—be composed of any character sequence. Quotation is used to define identifiers containing whitespace, brackets, or other reserved symbols. The same holds for string literals (*string*).

The descriptive power of M3L lies in the fact that the formal semantics is rather abstract. There is no fixed domain semantics connected to M3L definitions. The semantics of M3L evaluation will not be discussed in this paper. For more details see [9].

 ISBN: 978-1-61208-533-3 

### A. Concept Definitions and References

A M3L definition consists of a series of definitions (⟨*def*⟩ in the grammar definition above). Each definition starts with a previously unused identifier that is introduced by the definition and may end with a semicolon, e.g.:

```
NewConcept;
```

We call the entity referenced by such an identifier a *concept*.

The keyword `is` introduces an optional reference to a base concept. An inheritance relationship as known from object-oriented modeling is established between the base concept and the newly defined derived concept. This relationship leads to the concepts defined in the context (see below) of the base concept to be visible in the derived concept. Furthermore, the refined concept can be used wherever the base concept is expected (similar to subtype polymorphism).

As can be seen in the grammar, the keyword `is` always has to be followed by either `a`, `an`, or `the`. The keywords `a` and `an` are synonyms for indicating that a classification allows multiple sub concepts of the base concept:

```
NewConcept is an ExistingConcept;
NewerConcept is an ExistingConcept;
```

There may be more than one base concept. Base concepts can be enumerated in a comma-separated list:

```
NewConcept is an ExistingConcept,
              an AnotherExistingConcept;
```

The keyword `the` indicates a closed refinement: there may be only one refinement of the base concept (the currently defined one), e.g.:

```
TheOnlySubConcept is the SingletonConcept;
```

Any further refinement of the base concept(s) leads to the redefinition ("unbinding") of the existing refinements.

Statements about already existing concepts lead to their redefinition. E.g., the following expressions lead to the same definition of the concept `NewConcept` as the above variant:

```
NewConcept;
NewConcept is an ExistingConcept;
NewConcept is an AnotherExistingConcept;
```

### B. Content and Context Definitions

Concept definitions as introduced in the preceding section are valid in a *context*. Definitions like the ones seen so far add concepts the topmost of a tree of contexts. Curly brackets open a new context, e.g.:

```
Person { name is a String; }
Peter is a Person{"Peter Smith" is the name;}
Employee { salary is a Number; }
Programmer is an Employee;
PeterTheEmployee is a Peter, a Programmer {
 30000 is the salary; }
```

In this example, we assume that concepts `String` and `Number` are already defined. The subconcepts created in context are unique specializations in that context only. In practice, the concept `30000` should also be given. If not, it will be introduced locally in the context of `PeterTheEmployee`, preventing reuse of the identical number.

M3L has visibility rules that correlate to contexts. Each context defines a scope in which definition identifiers are valid. Concepts from outer contexts are visible in inner scopes. E.g., in the above example the concept `String` is visible in `Person` because it is defined in the topmost scope. `salary` is visible in `PeterTheEmployee` because it is defined in `Employee` and the context is inherited. `salary` is not valid in the topmost context and in `Peter`. Contexts with those names may be defined later on, though.

Tying a context to a concept can be interpreted in different ways, e.g., as contextualization or as aggregation.

Contexts can be referenced using the projection operator `from` in order to use concepts across contexts:

```
salary from Employee.
```

### C. Narrowing and Production Rules

M3L allows assigning one *semantic production rule* to each concept. Production rules fire when an instance comes into existence that matches the definition of the left-hand side of the rule. They replace the new concept by the concept referenced by the right-hand part of the rule.

The following shows an example:

```
Person {
 female is the sex; married is the status;
} |= Wife;
```

Whenever a `female Person` who is `married` shall be created then a `Wife` is created instead.

Production rules are usually used in conjunction with M3L's *narrowing* of concepts. Before a production rule is applied, a concept is narrowed down as much as possible. Narrowing is a kind of matchmaking process to apply the most specific definition possible.

If a base concept fulfills all definitions—base concepts and constituents of the context—of a derived concept, then the base concept is taken as an equivalent of that derived concept. If a production rule is defined for the derived concept, this rule is used in place of all production rules defined for any super concept.

The following code shows an example of combined narrowing and semantic production rules:

```
Person {
 sex; status; }
MarriedFemalePerson is a Person {
 female is the sex; married is the status;
} |= Wife;
MarriedMalePerson is a Person {
 male is the sex; married is the status;
} |= Husband;
```

There is a concept `Person`. Whenever an "instance" (a derived concept) of `Person` is created, it is checked whether it actually matches one of the more specific definitions. A married female `Person` is replaced by `Wife`, a married male `Person` by `Husband`, every other `Person` is kept as it is:

```
Person {                    → Person {
 male is the sex; }           male is the sex;}
Person {
 female is the sex;         → Wife;
 married is the status; }
Person {
 male is the sex;           → Husband;
 married is the status; }
```

In addition to the semantic production rules that create new concepts, M3L also has *syntactic production rules*:

```
Person { name is a String; }
 |- "<person>" name "</person>";
```

Syntactic production rules evaluate to a string. The rules consist of a list of string literals and concept references whose production rules are applied recursively.

The syntactic rules are also used as grammar rules to generate recognizers that create concepts from strings.

If no rule is given, then the default production rule evaluates a concept to its name.

## V. M3L FOR MULTILINGUAL CONTENT

To demonstrate how the M3L can be used to model multilingual content, we use a M3L representation of the setup from Fig. 1. Local models are derived from a central repository, and we briefly touch workflows and content interchange formats.

### A. Content Models

We use concepts to model repositories, local collections, and content as shown in Fig. 1. Their contextualization represents content structure. Relationships between repositories or collections are established by derivation.

For the example, we concentrate on the navigation structure. This frees us from content modeling details that are not relevant for the discussion.

We use contextualization for the navigation hierarchy:

```
GeneralRepository is a ContentRepository {
 GeneralContent is a Content { … }
 GeneralNavigation is a Navigation {
  "Products+Services" is a NavItem {
   "Consumer Products" is a NavItem;
   "Professional Products" is a NavItem;
   Support is a NavItem;
} } }
```

*ContentRepository*, *Content*, *Navigation*, and *NavItem* may be given concepts here.

In this example, the general repository hosts a central navigation structure with a main navigation node *Products+Services*. It has subordinate navigation items *Consumer Products*, *Professional Products*, and *Support*.

The following models use derivation to relate translations of navigation items to those in the general repository.

We present two modeling alternatives to translate the navigation hierarchy. In the first alternative, editors translate each navigation item one by one. This way, the structure is kept as it is. We do so by deriving a sub concept, e.g., *GermanNavigation* from the general navigation. In this "copy" of the general navigation we can locally "replace" the navigation items by translations.

```
GermanRepository is a GeneralRepository {
 GermanContent is the GeneralContent { … }
 GermanNavigation is the GeneralNavigation {
  Produkte+Dienste is the Products+Services {
   Verbraucher is the "Consumer Products";
   Profis is the "Professional Products";
   Kundendienst is the Support;
} } }
```

We provide exactly one translation (`is the`) per navigation item in the specific region context. In other contexts other translations can be given.

Changes in the general repository are propagated to local ones in such a model. E.g., when a new navigation item is added globally, it is inherited in the local repositories. Such an item will not be translated automatically, but the overall navigation structure stays up-to-date.

As a second alternative we create a navigation structure locally. We populate it by picking single instances from the general repository. This way we detach the local structures from the global structure. The other properties, e.g., the pages assigned to a navigation node, are inherited, though.

Building *GermanNavigation* this way results in:

```
GermanRepository is a Repository {
 GermanContent is a Content { … }
 GermanNavigation is a Navigation {
  Produkte+Dienste is a Products+Services
   from GeneralNavigation
    from GeneralRepository {
   Verbraucher is a "Consumer Products"
    from GeneralNavigation
     from GeneralRepository;
   Profis is a "Professional Products"
    from GeneralNavigation
     from GeneralRepository;
   Kundendienst is a Support
    from GeneralNavigation
     from GeneralRepository;
} } }
```

The repository base is a new, "empty" one since *GermanNavigation* is derived from just *Navigation*, not *GeneralNavigation*. The inserted navigation items are derived from those from the global repository, though.

In such a detached repository, possible changes in the central repository are not propagated, but have to be reapplied locally. This can be performed either completely manually, or by means of a workflow (see below).

When structures are changed during localization, there are various possibilities for structural differences. The following model gives two examples (without translation) for a company's web site in countries with smaller markets and, therefore, a smaller offering:

```
NicheMarkets is a ContentRepository {
 SmallCountry1 is a GeneralRepository {
  Country1Content is the GeneralContent { … }
  Country1Nav is the GeneralNavigation {
   Products is the Products+Services {
    "Consumer Products";
    "Professional Products";
 } } }
 SmallCountry2 is a GeneralRepository {
  Country2Content is the GeneralContent { … }
  Country2Nav is the GeneralNavigation {
   Products is the Products+Services; }
} }
```

In the first example, *SmallCountry1*, a subset (two out of the three) of navigation items is inserted into the navigation tree below *Products*, the navigation item that generally appears as *Products+Services*. The second example, *SmallCountry2*, shows a flatter structure with no sub navigation items under *Products*.

Content and its structure are localized the same way as the navigation structure. We limit the example to navigation items in order to reduce its complexity.

Along with the content also the layouts used for its publication can be localized when documents are produced using M3L's syntactic productions. E.g., the production rules can generate HTML pages. In a very simple way:

 ISBN: 978-1-61208-533-3 

```
GreekRepository is a GeneralRepository {
 GreekContent is the GeneralContent {
  GreekPage is a Page {…} |- … (Greek layout)
}
FrenchRepository is a GeneralRepository {
 FrenchContent is the GeneralContent {
  FrenchPage is a Page {…} |- … (French layout)
}
```

### B. Workflows

Typically, translation tasks are driven by workflows. Introducing a complete workflow management system is beyond the scope of this paper. We provide a sketch of an approach based on M3L structures. A workflow consists of workflow tasks, e.g., represented by derivations of:

```
WorkflowTask is a … { Agent is a …; }
```

Then a translation workflow task may look like:

```
TranslationWorkflowTask is a WorkflowTask {
 ContentToTranslate is a String;
 ResultingContent is a String;
 Translator is the Agent;
} |= TranslatedContent;
```

For the sake of simplicity we assume content to consist just of *String*s.

When the task is completed, it evaluates to a *TranslatedContent*. We need this type to distinguish it from *InternationalizedContent* (s.b.) that is processed in workflows.

We connect content to workflows by means of semantic production rules. E.g., the following example shows a definition of content of type *News* with a production rule creating a workflow task.

```
News is a TranslatedContent;
GeneralNews is an InternationalizedContent {
 Title is a String; Text is a String;
} |= TranslationWorkflowTask {
 Title is a ContentToTranslate;
 Text is a ContentToTranslate;
 … Translator;
} |= News { … };
```

Therefore, whenever new content that is derived from *GeneralNews* is created, the rule is inherited and thus a *TranslationWorkflowTask* is created, initialized with the *News*' *Title* and *Text* as content that needs translation. It yields the translated *News*.

### C. Content Exchange

In manual translation processes, content needs to be shipped between different parties.

Inside one organization, communication can be established using M3L's structures directly. In order to interchange content with external organizations, we use an external format for input and output. This can be defined using M3L's syntactic production rules. Example:

```
Content News { … } |- "<xliff …> … <source>"
                      Text "</source> … </xliff>"
```

Here, the *Text* component of content of type *News* is externalized in XLIFF. The resulting file can be sent to a translator, and the result can be parsed in to form a *News*.

## VI. Summary and Outlook

This section recaps the paper and discusses future work.

### A. Summary

Multilingual content management is in widespread use, and requirements for the management of localized variants of global content can be formulated. This paper discusses an approach to multilingual content management using context.

The Minimalistic Meta Modeling Language (M3L) is a general-purpose modeling language that has proven particularly useful for context-aware content management. In this paper we demonstrate how to employ M3L to model multilingual content management in a product-agnostic way.

### B. Outlook

M3L can be executed by evaluating M3L statements. However, this execution is not an adequate approach for building running systems. CMS products, on the other hand, are of practical importance. Therefore, in the future we want to build product-specific model compilers that generate product configurations out of M3L statements.

The workflows for content localization need further work, in particular those incorporating external translators.

## Acknowledgment

The author wishes to express gratitude to his employer, Namics, for enabling him to follow his scientific ambitions.

The insights presented here are taken from numerous practical projects. Thanks to all the colleagues as well as the business and technology partners that helped understanding problems and developing ideas for their solution.

## References

[1] S. Mescan, "Why Content Management Should Be Part of Every Organization's Global Strategy," Information Management Journal, vol. 38, no. 4, pp. 54-57, Jul./Aug.2004.

[2] S. Huang and S. Tilley, "Issues of Content and Structure for a Multilingual Web Site," Proc. 19th Annual International Conference on Computer Documentation (SIGDOC '01), ACM New York, NY, USA, pp. 103-110, Oct. 2001.

[3] R. Lockwood, "Have Brand & Will Travell," Language International, vol. 12, no. 2, pp. 14-16, 2000.

[4] J.-M. Lecarpentier, C. Bazin, and H. Le Crosnier, "Multilingual Composite Document Management Framework For The Internet: an FRBR approach," Proc. 10th ACM Symposium on Document Engineering (DocEng '10), ACM New York, NY, USA, pp. 13-16, Sep. 2010.

[5] P. Sandrini, "Website Localization and Translation," Proc. EU High Level Scientific Conferences, Marie Curie Euroconferences, MuTra: Challenges of Multidimensional Translation, pp. 131-138, May 2005.

[6] D. Jones, A. O'Connor, Y. M. Abgaz, and D. Lewis, "A Semantic Model for Integrated Content Management, Localisation and Language Technology Processing," Proc. 2nd International Conference on Multilingual Semantic Web (MSW'11), vol. 775, pp. 38-49, 2011.

[7] R. Miller, "Multilingual Content Management: Found in Translation," EContent, vol. 29, no. 6, pp. 22-27, Jul. 2006.

[8] P. Tonella, F. Ricca, E. Pianta, and C. Girardi, "Restructuring Multilingual Web Sites," Proc. International Conference on Software Maintenance, pp. 290-299, Oct. 2002.

[9] H.-W. Sehring, "Content Modeling Based on Concepts in Contexts," Proc. The Third International Conference on Creative Content Technologies (CONTENT 2011) , pp. 18-23, Sep. 2011.

 ISBN: 978-1-61208-533-3 

# Video Fingerprinting by Common Features in a Scene

Jongweon Kim*, Sungjun Han**, Yongbae Kim**
*Dept. of Contents and Copyright, **Creative Content Labs
Sangmyung University
Seoul, Korea
Email: jwkim@smu.ac.kr, sungjun@cclabs.kr,
ybkim@cclabs.kr

Jungjae Lee***
***Dept. of Entertainment Business
Soongsil Cyber University
Seoul, Korea
Email:jjlee@mail.kcu.ac

***Abstract*—Video fingerprinting is an important aspect in the copyright protection field, as digital environment enables the copyright infringement to get easier and easier. There are many video contents on the Internet. Copyright owners want to identify contents on the net and to block infringed contents. The video content is invaluable because the owners invested a huge amount of money when they made the movie. In this paper, we propose an efficient algorithm to identify video contents even if we only have a video frame. The algorithm divides a video content into scenes and then extracts common features from a scene. The feature database contains only a set of features per scene. That means the proposed algorithm optimizes the feature database and the time it takes to compare the features. Also, this algorithm can identify a video using only a frame of the video.**

***Keywords-video identification; common feature; scene; scale invariant feature transform .***

## I. INTRODUCTION

Video fingerprinting technology is an important aspect related to the video search and copyright protection. In the copyright protection field, content fingerprinting is a technical measure to block the illegal distribution of copyrighted contents on the net. Generally, there are three kinds of content filtering, namely keyword-based, hash-based, feature-based filtering. The feature-based filtering is the most powerful technology to identify the contents under several distortion attacks.

Digital Rights Management (DRM) is the most secure measure [1] to protect the content because it uses encryption technology. In February of 2007, Steve Jobs, who was former CEO of Apple Inc., announced the introduction of DRM-free service [2]; next, DRM started to disappear from contents market. Digital watermarking technology [3][4] is an alternative to the DRM-free service, but the watermark information should be embedded into the content before distribution. Indeed, the digital watermarking is not perfect to protect contents and it is sensitive to malicious attacks. These are drawbacks of the digital watermarking technology.

Watermarks offer some advantages over fingerprinting. A unique watermark can be added to the content at any stage in the distribution process and multiple independent watermarks can be inserted into the same video content. This can be particularly useful in tracing the history of video copies. Detecting watermarks in a video can indicate the source of an unauthorized copy.

While video fingerprinting systems must search a potentially large database of reference fingerprints, a watermark detection system only has to do the computation to detect the watermark. This computation can be significant and, when multiple watermark keys must be tested then, watermarking can fail to scale to User Generated Content (UGC) site volumes.

Fingerprinting technology has a number of advantages over the conventional DRMs and digital watermarking technologies. Fingerprinting technology does not need to embed any information before distribution and just stores the features into database in contrast with digital watermarking. The key problem of DRM is interoperability among DRMs and the fingerprinting enables the authorized users to use the content without any barriers. Fig. 1 shows the use case of the fingerprinting technology for the copyright protection.

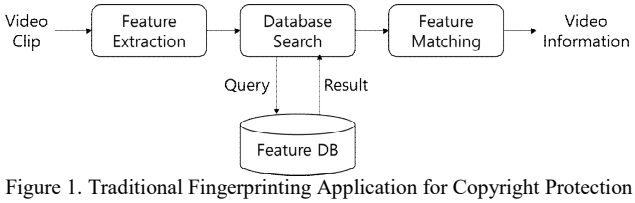



Figure 1. Traditional Fingerprinting Application for Copyright Protection

Normally, digital contents are compared based on hash values that are directly derived from the digital components of a content. However, such methods are incomplete as they can only determine absolute equality or non-equality of video data files or parts. More often than not, differences in a video codec and digital processing artifacts may cause small differences in the digital components without changing the video perceptually. Thus, when employing hash methods, a comparison for absolute equality may fail even when two video clips are perceptually identical. Moreover, hash based filtering is also of little value when one wishes to identify video clips that are similar (but not identical) to a given reference clip. The limitation of the equality and inequality decision by hash value is that the hash-based technique is not available for the similar searching [5].

On the other hand, video fingerprinting technique enables identification of videos with a different resolution compared with the original (smaller or larger) as well as identifying videos that have been slightly modified (blurring, rotation,

 ISBN: 978-1-61208-533-3 

acceleration or deceleration, cropping, insertions of new elements in the video), and videos where the audio track has been modified [5].

For the video fingerprinting, Mani Malek Esmaeili et al. suggested a fast video fingerprinting technology [6], Bo Wu et al. proposed a robust video fingerprinting using sparse represented features [7] and Mu Li et al. proposed a compact video fingerprinting using structural graphical model [8]. Although their algorithms show good performance to identify a video, the algorithms require several frames or scenes.

There are many fingerprinting algorithms in the image processing field. Nowadays, the Scale Invariant Feature Transform (SIFT) [9] algorithm is powerful to extract features from images. Although SIFT is the most powerful feature extraction algorithm, SIFT is inefficient for extracting features from video because it is a complicated algorithm and it occupies many computational resources.

In this paper, we propose a new method to block contents using fingerprinting. The rest of this paper is organized as follows. Section II describes the theory of SIFT and the burdens of the algorithm. Section III describes the proposed method and its procedure. Section IV addresses the experiment and results. Section V concludes the paper.

## II. SCALE INVARIANT FEATURE TRANSFORM

The SIFT can extract image features that are invariant to scale and rotation. The SIFT algorithm is comprised of four main stages: scale space extrema detection, keypoint localization, orientation computation and keypoint descriptor extraction.

The first stage to detect scale space extrema is the process to detect the invariant interest point using Difference of Gaussians (DoG) to identify the potential keypoints which are extrema. DoG is an approximation of Laplacian of Gaussians (LoG) and has low computational complexity. The Gaussian blurred images at six different scales are produced from the input image and DoGs are computed from neighbors to extract local extrema in scale space. In the second stage for keypoint localization, candidates of keypoint are localized by detecting extrema in the DoG images that are locally extremal in space and scale. The unstable kepoints (usually edges) in space are removed by thresholding for the ratio of eigenvalues of the Hessian matrix (unstable edge keypoints have high ratios, and stable corner keypoints have low ratios), low contrast keypoints are removed and the remaining keypoints are localized by interpolating across the DoG images. The third stage for orientation computation is the process to assign a principal orientation of keypoint. The directions of pixels around keypoint are computed and the histogram of the directions is used to select the orientation of keypoints. If there is another orientation over 80% of maximum histogram, the stage assigns additional keypoint. This means there can be one or more keypoints at the same point. The final stage computes the orientation of the gradients around a keypoint. This is the stage to make a highly distinctive descriptor for each keypoint. For the orientation invariance, the descriptor coordinates and gradient orientations are rotated relative to the orientation of keypoint.

For every keypoint, a set of orientation histograms is created on 4x4 pixel neighborhoods with 8 bins each (using magnitudes and orientation of samples in 16 x 16 region around the keypoint). The resulting feature descriptor will be a vector of 128 elements that is then normalized to unit length to handle illumination differences. Descriptor size can be varied, however best results are reported with 128D SIFT descriptors. SIFT descriptors are invariant to rotation, scale, contrast and partially invariant to other transformations. The SIFT descriptor size is controlled by its width, i.e. the array of orientation histograms (n x n) and number of orientation bins in each histogram (r). The size of resulting SIFT descriptor is $rn^2$. The value of n affects the window size around the keypoint as we use 4 x 4 region to capture pattern information, e.g. for n = 3, we will use a window, size of 12 x 12 around the keypoint. Various sizes were analyzed in [10] and it was reported that 128D SIFT is superior in terms of matching precision, i.e. n = 4 and r = 8. Most other works have used standard 128D SIFT features while very few have tried smaller SIFT descriptors for small scale works, e.g. 36D SIFT features from 3 x 3 subregions, each with 4 orientation bins, with few target images are used in [11].

Smaller sized descriptors use less memory and result in faster classification but precision rates may be affected. No research article has investigated the classification performance of SIFT descriptors of size other than 128.

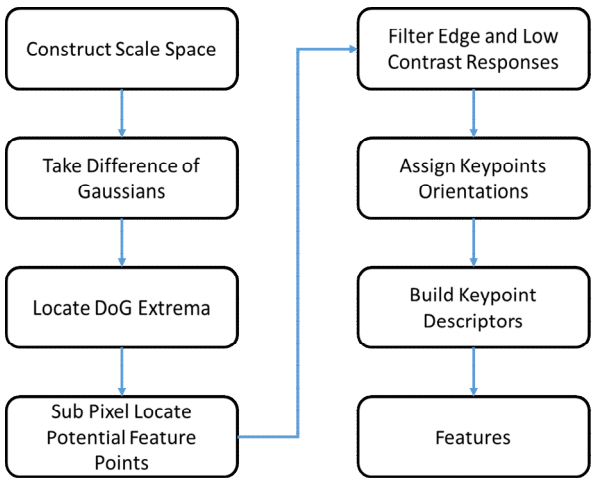



Figure 2. Procedure of SIFT

Video clips have 30 frames per second. If SIFT is applied to video content for the video fingerprinting, the algorithm should process to extract features from each frame. This means the computational amount is very high and it is not useful for video identification.

## III. PROPOSED METHOD

Our goal of the paper is how to identify the video from a frame. For this purpose, we developed a simple scene detection algorithm and a video fingerprinting technology using SIFT which has the low complexity for the image

 ISBN: 978-1-61208-533-3 

identification. First of all, we have to improve the computational complexity before using SIFT as a video fingerprinting technology. There are some candidates to reduce the computational amount of the feature extraction. One candidate is a temporal feature extraction from video clips. Although the temporal feature has low computational complexity, this feature cannot distinguish the video clips frame by frame. The other candidate is the binary feature extraction which is proposed by Lee et al [12]. The binary fingerprints are obtained by filtering and quantizing intermediate features extracted from an input video clip. The filters and their associated quantizers for the fingerprint extraction are selected from a class of candidate filters and quantizers using the Symmetric Pairwise Boosting (SPB) algorithm.

Our approach is to reduce the computational amount for video clips when extracting the features. Even if the proposed algorithm reduces the computational operation, it can also identify the video clips by only a frame.

### A. *Scene Segmentation*

A video program such as motion pictures, TV movies, etc., has a story structure and organization. As illustrated in Fig. 3, three levels define this syntactic and semantic story structure: narrative sequence, scene and camera shot. A camera shot is a set of continuous frames representing a continuous action in time or space. It represents the fundamental unit of video production, reflecting a basic fragment of story units. A scene is a dramatic unit composed of a single or several shots. It usually takes place in a continuous time period, in the same setting, and involves the same characters. At a higher level, we have the narrative sequence, which is a dramatic unit composed of several scenes all linked together by their emotional and narrative momentum [13].

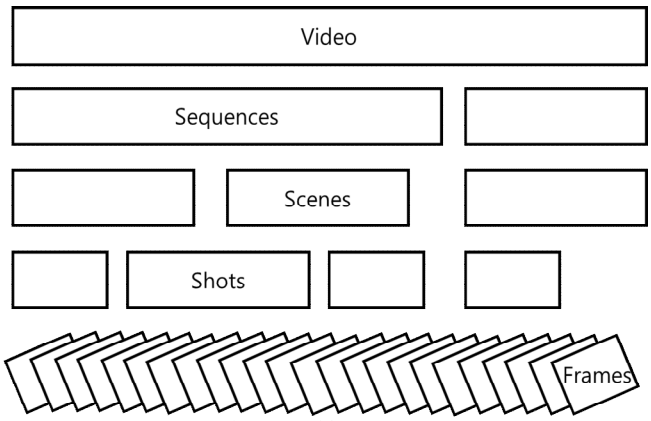



Figure 3. Video structure

In the proposed algorithm, the scene segmentation is achieved by using average of difference between frames. In a scene, the difference value of the consecutive frames is lower than that of the consecutive frames between scenes. This algorithm has simple architecture and computational advantage. Especially, this method is efficient to segment similar frames as a scene.

### B. *Features for Video Fingerprinting*

There are many common features between frames in a scene. The implementation process of the feature database is as follows:

Step 1: Segment the scene from video clips.
Step 2: Extract features from each frame.
Step 3: Choose common features from features of frames.
Step 4: Store common features into database.

In step 3, all features of a scene are arranged by same feature values and descriptors. By the frequency of the same feature in a scene, the features are sorted and then selected as the common feature.

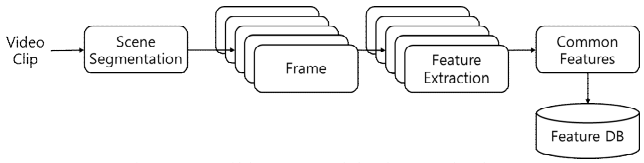



Figure 4. Build process of the feature database

Fig. 4 shows the build process of the feature database.

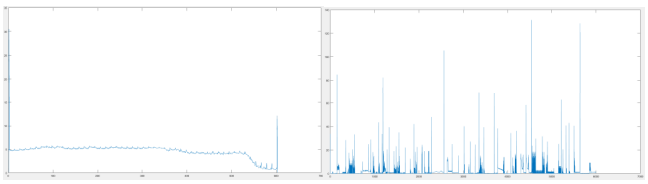
Once the feature database is implemented, the identification process is as follows:

Step 1: Choose a frame from video clips.
Step 2: Extract features from the selected frame.
Step 3: Compare the features with the database.
Step 4: Identify the video.

The main idea of the proposed algorithm is using the common features among the video scene. Any one of the fingerprinting algorithms, such as SIFT, SURF, etc., can be used to extract common features.

## IV. EXPERIMENTS AND RESULTS

To evaluate the proposed method for the video fingerprinting, we have taken 4 video clips. At the first step, the video clips have been segmented by scenes and we extracted the features from each frame of the scene. The features from each frame of the scene are refined as common features between frames.



(a) Sample video clip 1 (b) Sample video clip 2

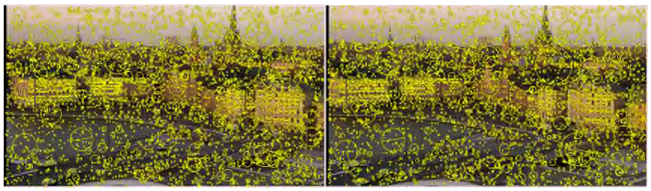
Figure 5. Scene segmentation graphs

Fig. 5 depicts the scene segmentation results of the test video clips. The pulses in the graph are boundaries of the

scene. Graph (a) shows the video clip 1 has only a scene, which is slightly changed between the frames in the scene. Video clip 2 has many scenes, as shown in graph (b).

If the interval between the peaks is long, there is a long scene and if the interval is short, there is a short scene. The graph for scene segmentation is calculated from average value of the difference between a frame and the neighbor frame.

After scene segmentation, the features of every frame in the scene are extracted and then we choose the common features of all frames. Fig. 6 shows the extracted features from frames and the common features. (a) shows the features from the first frame of a scene in the test video clip 1 and the 16th frame of same scene. (b) shows the common features among all frames of the scene in the clip 1. The yellow circles are features which are extracted by SIFT. The different circle size means the feature is extracted in the different scale domain and the line in the circle represents the orientation of the feature.



(a) Features from frame 1 and 16 of a scene in clip 1

(b) Common features of a scene in clip 1

Figure 6. Features from each frame and common features

## V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a video fingerprinting method to identify video clips using only a frame. The video fingerprinting technology can block the illegal distribution of the infringed video contents on the internet. Our approach used spatial features of each frames and reduced the size of the feature database and amount of features in a scene. For achieving this purpose, we segmented the video clips into scenes, extracted the features of each frame in a scene and chose the common features in a scene. As a result, the number of the common features is less than the average number of the features of each frame. This results in less computation complexity when the video fingerprinting is applied to filtering of infringed contents. Moreover, the approach can identify the video clip even if there is only a frame of the video clip.

In the future work, we are going to improve the identification speed and to develop a common feature extraction algorithm. We are also planning to study a fast algorithm for comparison search in feature database.

## ACKNOWLEDGMENT

This work was supported by the Technological Innovation R&D Program (S2380813) funded by the Small and Medium Business Administration (SMBA, Korea)"

## REFERENCES

[1] Digital Rights Management, Wikipedia, visited Oct. 28th 2016. https://en.wikipedia.org/wiki/Digital_rights_management

[2] Steve Jobs, Thoughts on Music, Apple Wet Site, Feb., 2007. http://www.apple.com/kr/hotnews/thoughtsonmusic/

[3] I. J. Cox, J. Kilian, T. Leighton and T. Shamoon. Secure Spread Spectrum Watermarking for Multimedia. IEEE Transactions on Image Processing, vol. 6, pp.1673-1687, 1997.M

[4] J. H. Nah, J. W. Kim and J. S. Kim, Video Forensic Marking Algorithm using Peak Position Modulation, Appl. Math. Inf. Sci., vol.7, no.6, pp.2391-2396, 2013.

[5] Digital video fingerprinting, Wikipedia, visited Oct., 25, 2016, https://en.wikipedia.org/wiki/Digital_video_fingerprinting

[6] M. M. Esmaeili, M. Fatourechi, and R. K. Ward, A Robust and Fast Video Copy Detection System Using Content-Based Fingerprinting, IEEE Transactions on Information Forensics and Security, vol. 6, no. 1, pp.213-226, 2011

[7] B. Wu, S. Krishnan, N. Zhang and L. Su, Compact and Robust Video Fingerprinting using Sparse Represented Features, Multimedia and Expo (ICME), 2016 IEEE International Conference on, pp.1-6, 2016

[8] M. Li and V. Monga, Compact Video Fingerprinting via Structural Graphical Models, IEEE Transactions on Information Forensics and Security, vol.8, no.11, pp.1709-1721, 2013

[9] N. Y. Khan, B. McCane and G, Wyvill, " SIFT and SURF Performance Evaluation Against Various Image Deformations on Benchmark Dataset", International Conference on Digital Image Computing: Techniques and Applications, pp.501-506, 2011.

[10] D. Lowe, Distinctive Image features from scale invariant keypoints, International journal of Computer Vision, 60, pp.91-110, 2004.

[11] W. Daniel, R. Gerhard, M. Alessandro, D. Tom and S. Dieter, Pose Tracking from Natural Features on Mobile Phones, Proc. International Symposium on Mixed and Augmented Reality, pp.125-134, 2008.

[12] S. Lee, C. D. Yoo and T. Kalker, "Robust Video Fingerprinting Based on Symmetric Pairwise Boosting," IEEE Transactions on Circuit and Systems for Video Technology, vol.19, no.9, pp.1379-1388, Sep. 2009, doi: 10.1109/TCSVT.2009.2022801

[13] W. Mahdi, L. Chen and M. Ardebilian, Automatic video scene segmentation based on spatial-temporal clues and rhythm, 2014. https://arxiv.org/abs/1412.4470v1

 ISBN: 978-1-61208-533-3 

# A View Synthesis Approach for Free-navigation TV Applications

Ilya Ganelin
Electrical & Computer Engineering Department, University of British Columbia
ICICS, University of British Columbia
Vancouver, BC, Canada
iganelin@ece.ubc.ca

Panos Nasiopoulos
Electrical & Computer Engineering Department, University of British Columbia
ICICS, University of British Columbia
Vancouver, BC, Canada
panos@ece.ubc.ca

Mahsa T. Pourazad
TELUS Communications Incorporation
Vancouver, Canada,
pourazad@ece.ubc.ca

***Abstract*—The need for multiview content is more pronounced with the emergence of Free-viewpoint Television (FTV), Super Multiview (SMV), and Free Navigation (FN) technologies. For multiview content creation, it is not practical to capture all of the views required by different multiview display technologies. Instead, a limited number of views are captured and the remaining views are synthesized using the available views. The efficiency of the view synthesizing process has a high impact on the quality of the generated multiview content. In this paper, we present a novel view synthesizing scheme, which utilizes unique techniques such as background decomposition in three layers, background edge dilation, vertical interpolation and edge aware warping, to generate high quality virtual views. Subjective evaluations confirm that our approach outperforms the state-of-the-art interpolation-based view synthesizing.**

***Keywords-Free Navigation TV; Multiview TV; view synthesis; hole filling; Multimedia communication; Image generation; Image reconstruction.***

## I. INTRODUCTION

Free-viewpoint Television (FTV) provides the viewers with realistic impression of a scene by allowing them to freely navigate through the scene in Free Navigation (FN) applications or perceive scene depth in the case of Super Multiview (SMV) applications [1]. There are a number of hurdles in the proliferation of these technologies, such as availability, production, and transmission of multiview content to the end user. Multiview content production is expensive and highly demanding in terms of camera configuration and post processing [1]. As FTV technology evolves, manufacturers attempt to provide viewers with a larger number of views to improve transition between sweet spots. As a result, the number of views of the preliminary multiview content will no longer be enough, and thus synthesizing virtual views becomes essential. In the case of FN, where captured views are further apart, the quality of the synthesized view is even more important as there is no additional information from a neighboring view as in the case of SMV.

The main challenge with view synthesis is estimating the information of the occluded areas [1]. A common solution is to apply inter pixel interpolation to estimate the missing texture. This approach has been utilized in the state-of-the-art View Synthesis Reference Software (VSRS) [4], which has been adopted by the MPEG-3DV group to synthesize test sequences for 3D video compression standardization activities [2][4]. Unfortunately, the downfall of all interpolation-based hole-filling methods is that the interpolated texture does not resemble the true structure of the occluded areas.

In a different approach presented in [3], warping is utilized for synthesizing additional views. In this method, first a sparse saliency map is created. This map helps to separate foreground from background and then the saliency map information is used to stretch background parts of the picture and cover occluded areas. This technique minimizes the visible distortions of the image, but due to stretching it can also distort some human visual cues such as vertical lines and shapes of known objects, e.g., faces.

To overcome shortcomings of the existing schemes, we propose a new and unique view synthesizing scheme which uses background-to-foreground warping and background separation for accurate filling of occluded regions. Our method improves on the inter pixel interpolation idea of the VSRS approach as well as the Disney's warping technique [2][3][4]. The performance of the proposed technique is compared with that of the state-of-the-art VSRS [2][4], for both view extrapolation and view interpolation scenarios (see Scheme below) subjectively.

The remainder of this paper is structured as follows: Section 2 describes our method, Section 3 presents the experimental results, and conclusions are drawn in Section 4.

 ISBN: 978-1-61208-533-3 

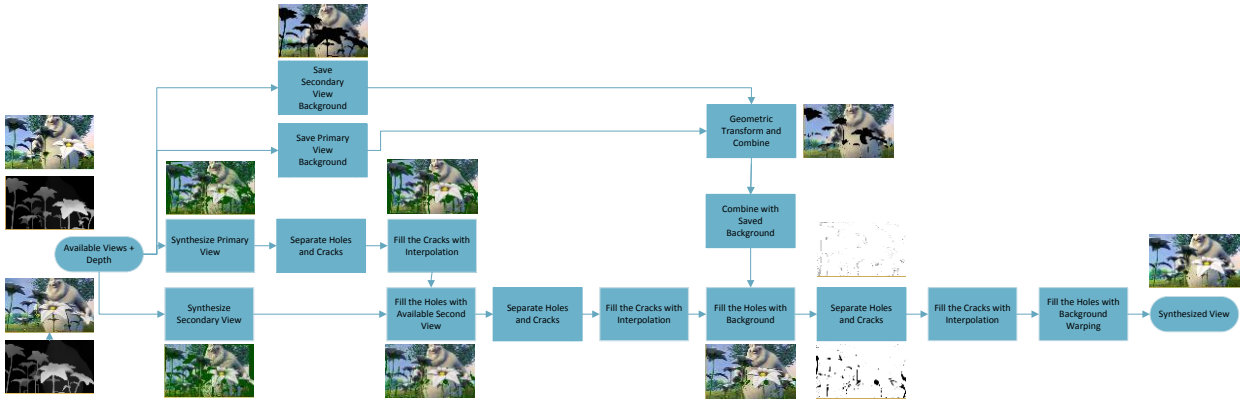



Figure 1: Block diagram of our method

## II. OUR VIEW SYNTHESIS APPROACH

### A. *Solving background leakage*

For FTV applications, it is common practice that several different views are captured with multiple cameras (usually in a parallel or arch setup) and additional views are synthesized, as if there were more cameras in the original setup. The resulting original views are spread apart from each other, so that many views need to be synthesized in between to generate free navigation or SMV (3D) content. Figure 1 shows the block diagram of our approach and the different stages of our hole filling process. In our approach, we create a primary synthesized view using the closest available real camera view position to the location of the synthesized view.

The translation process of pixels from the original view to the synthesized one creates holes (pixels with missing pixel values) in a way similar to a disparity-based synthesizing approach [4].

Due to the fact that the depth map of a foreground object might have different values, because of its volumetric nature, transitioning these pixels from one camera plane to another may result in consecutive pixels of the same object being space-separated. The background pixels might be shifted into these spaces and no hole would be identified in these locations. Therefore, these "empty" locations will not be included in the hole filling process. This effect is called background leakage. Figure 2a shows the leakage caused by the VSRS approach (artifacts on the flower). In our approach, in order to address this issue we separate each frame into three non-overlapping depth layers using two threshold values. We start the transformation of the pixels to the virtual camera plain from the foreground layer, followed by middle ground, and, lastly, the background. We limit the pixel transition such that the pixels from lower layers cannot be shifted within the upper layer object's boundaries. As a result, we do not end up with the background leaking to the occluded areas (Figure 2b). The next step in our process involves filling the occluded areas.

### B. *Holes Filling Using Interpolation*

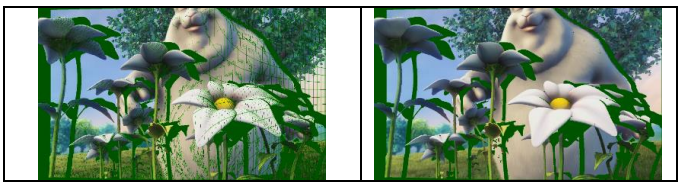
At the beginning of this stage, the occluded regions are classified as cracks (region sizes less than 0.3% of the frames width) and holes (rest of the occluded regions). The cracks are filled using nearest neighbor interpolation (similar to the VSRS approach) as described by following equations 3 and 4:

(a) (b)

Figure 2: Background Leakage in (a) VSRS, (b) our method.

$$Horizontal: new\_image(x, y) = image(x - 1, y) \quad (3)$$
$$Vertical: new\_image(x, y) = image(x, y - 1) \quad (4)$$

Figure 4a shows the primary synthesized view with cracks and holes shown in green. Figure 4b shows the same image after the cracks are filled by interpolation. A secondary synthesized view is generated using the further away captured view in a similar manner. At this stage, information from the secondary view, if available, is used to fill the existing holes in the primary synthesized view. As a result, some holes may be completely filled and some others may become smaller falling into the previously defined crack

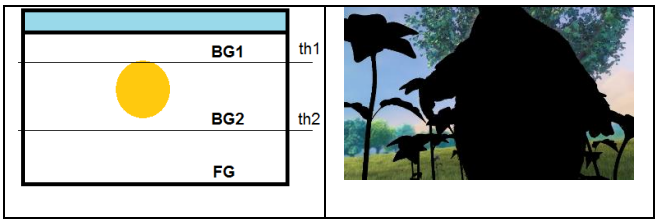


(a) (b)

Figure 3: (a) Primary synthesized view with cracks and holes shown in green; (b) cracks are filled by interpolation.

 ISBN: 978-1-61208-533-3 

Figure 4: Image resulted after the remaining holes were filled using information from the secondary synthesized view.




(a) (b)

Figure 5: (a) Simple Parallel Background (BG) Composition, (b) background 1 of the scene using simple parallel BG.

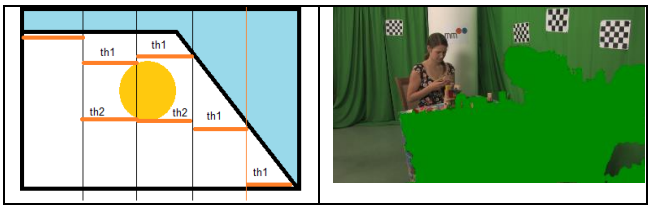
category. Figure 4 shows the resulting frame. Unlike VSRS, which fills all the holes using interpolation, we use temporal background information to partially fill the remaining holes.

### *C. Hole Filling Using Background Information*

As background tends to remain unchanged within a scene, we decided to use it for filling holes at this stage. Background separation is a challenging task, since defining what is background, depends on the scene composition and the subjective opinion of the viewer.

A simple approach is to define a single threshold based on the depth map of the whole scene. This approach can successfully handle the majority of outdoor scenes where the background is at the horizon and parallel to the camera's plane and the foreground objects are much closer to the viewer than the background. A more accurate approach, which we chose for our implementation, is Otsu's method, which chooses the threshold to minimize the intra class variance of the black and white pixels of the provided depth map [10]. Figure 5a shows the two thresholds (th1 and th2) used to define background 1, background 2 while Figure 5b shows background 1 for the outdoor scene with all the objects in front of th1 removed (dark regions).

For more complex, usually indoor scenes, the background may not be parallel to the camera's plane and its distance can vary from frame to frame (such as the green curtain in the Poznan Blocks sequence, Figure 6b). For such scenes we developed a different approach, where we separate each frame into five vertical slices, as shown in Figure 6a, compute the depth thresholds for each of them using the same Otsu's method, and merge them into a single




(a) (b)

Figure 6: (a) Slicing Complex Background into 5 vertical regions, (b) resulting background using vertical slicing.

Figure 7: (a) Misalignment of the color image and depth map (b) part of the hair on the right of the hole is identified as a background.

background image (image shown in Figure 6b). The reason for choosing five slices for the frame was the fact that the average width of the foreground objects in our test set was approximately one fifth of the frame's width (Figure 6a).

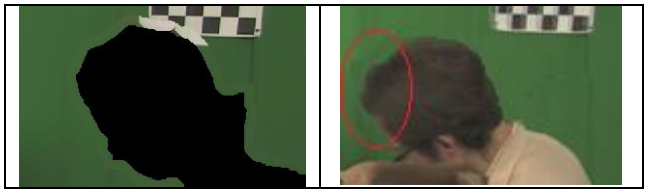
The complex scene approach can be used for all sequences, but since it is computationally more demanding, an automated classification into "simple" and "complex" background scenes is preferable.

For all concurrent frames we separated the background with the mentioned above approach and used SURF [5] to geometrically translate previously saved background image to the current camera's physical plane. After translation we filled the holes in the saved background image with the new available information from newly extracted background image, effectively increasing the coverage for future hole filling process.

There are cases where due to inaccuracies of the depth map, the foreground object's edges in the depth map are not aligned with those in the color image (Figure 7a), leading to various unwanted artifacts such as edge deformation of the foreground body (Figure 7b). Figure 7b shows how this mismatch will end up with object information on the other side of the hole as part of the background (see circled area which is part of the hair). Any effort to use the background information to interpolate or warp in order to fill the hole will result in foreground object information to be used as a "fill" for the hole as shown in Figure 8. In the interest of reducing edge artifacts and increasing the background coverage for further hole filling, we found through the tests that deleting 5 edge pixels from the background image (see Figure 9a) and then interpolating pixels from the holes' edges using 16 background pixels, effectively increases background coverage as shown in Figure 9b.

 ISBN: 978-1-61208-533-3 

Figure 8: Hole's Edge Deformation



(a) (b)

Figure 9: (a) Shows our background interpolation step expanding the hole and (b) shows the resulting artifacts using VSRS.

We take advantage of temporal redundancies in the background by tracking all the frames in a scene and identifying newly exposed background areas due to movement of foreground objects. As one would expect, this "extending" background allows us to cover more holes for the later frames, and it is efficient if we have relatively constant background and moving foreground objects.

The next stage involves separating the remaining occluded regions into cracks and holes once more and filling the cracks with the previously described interpolation process. The remaining holes are filled using the warping method presented in [1].

### *D. Warping*

The main difference in our warping process is the use of the edge mask. Since the human visual system is very sensitive to the vertical line distortions, we used the mask in order to stop the warping from deforming these lines. In order to compute the background edge mask we used Sobel edge detector on Luma component of the YUV frame. The warping process does not warp the background pass the detected edges and thus not destroying the background structure.

The second difference in our approach from the aforementioned one, is that in our implementation we warp/use background information that corresponds only to 85% of the hole's width, resulting in a better visual quality as we avoid excessive background warping. In the case where background is not available, such as the regions close to the edge of the image, we interpolate existing pixel values to fill in the hole.

### *E. Second View Interpolation*

As already mentioned, for view interpolation we use information from the second closest from the virtual camera position view was utilized as most reliable technique for holes filling, since it contains information from the real camera view. To this end, a secondary synthesized view was generated solely based on the further view by following the same procedure as creation of the primary synthesized view. Once the secondary synthesized view is generated, the holes in the primary synthesized view are filled by corresponding available areas in the secondary synthesized view. In order to make sure that the objects closer to the new camera plane are not obscured by background, the hole filling is performed from background towards foreground (using depth map information).

(a) (b)

Figure 10: (a) Final image synthesized using VSRS, (b) final image using our method that correctly copies foreground information from the second view.

As we can see from Figure 10a, in the case of VSRS the background information from second real view was copied over the foreground leave. Figure 10b shows the result of our method, where the flower and the leaves are complete, and there is no ghosting effect on the rabbit's hand.

## III. METHODOLOGY AND TEST RESULTS

To evaluate the performance of our algorithm, we conduct subjective tests and compare our synthesized views with those generated by the state-of-the-art VSRS [4]. The full paired comparison evaluation methodology is used for

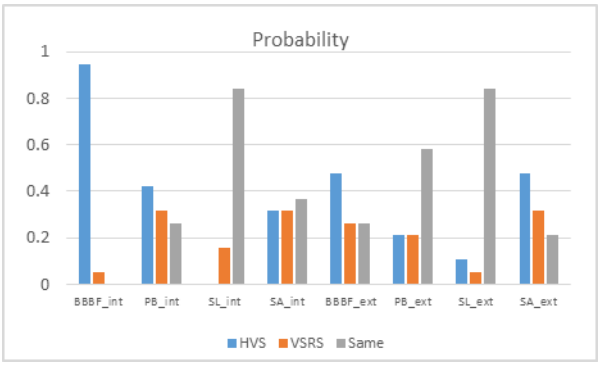



Figure 11: Probability for one of the method in each sequence to be chosen by the viewer.

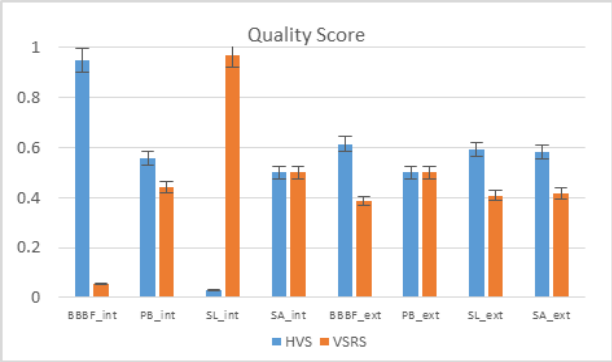



Figure 12: Quality Scores

our subjective tests [8]. A pair of the images is compared, with the subject asked to choose if either the "Left" or "Right" image is of better quality, or both are the "Same". For this evaluation we use four sequences recommended by MPEG [9]: "Soccer Linear2", "Soccer Arch1", "Poznan Blocks", and "Big Buck Bunny Flowers". We synthesize the required number of the virtual views between the provided real ones at the specified virtual points in space according to [9] using our approach and VSRS. We synthesized views using the two closest real views in the case of interpolation and using single closest view for extrapolation as the most appropriate case for FN. All sequences have Arch camera arrangement with different angle of convergence to the scene, except "Soccer Linear 2", which has linear camera arrangement. 19 subjects participated in the test. All the subjects are screened for the color blindness and vision acuity (Snellen and Ishihara charts) before conducting the test. Also to make them familiar with the test process, there was a training session using two test sequences ("Balloons" and "LoveBird1" [9]). After collecting test results, outliers were detected using circular triads method with defined threshold [8].

We use the Bradley-Terry model (BT) [7] combined with the maximum likelihood criterion as described in [8] to convert the results into the quality score metric. The pair ties are incorporated where they are available [6].

Figures 11 and 12 illustrate the subjective test results of our proposed method with those of VSRS for interpolation (marked as "int" for interpolation and "ext" for extrapolation in Figures 11 and 12) and extrapolation for the test sequences with 95% confidence interval.

As it can be observed, the "Big Buck Bunny" (BBBF) sequence shows significant improvement in both extrapolation and interpolation tests. The main reason for that is the fact that this sequence has color image perfectly aligned with the generated depth map and that the movement of the flowers and the rabbit exposed additional background that was stored and later used for the holes filling. The temporal background hole filling process bundled with the threshold based view synthesis handles this very well, improving overall quality of the synthesized view.

The "Poznan Block video" (PB) sequence shows small improvement, due to the overall low quality depth map, that does not align with the color image. "Soccer Arch's" (SA) modest gain, on the other hand, comes from the fact that the cameras' locations were far away apart and the camera calibration parameters were off. The misalignment of the left and right views is obvious on the synthesized views, making it a hard task to fill in the large holes. Our warping technic helps to slightly improve over VSRS.

In the case of "Soccer Line2" (SL), although it looks like the videos have completely different quality score, the results a priori show no statistically significant preference for HVS or VSRS, as the "same" option was selected in 84% of the cases (Figure 11). Video produced by our method, does not show any significant improvement over VSRS, since the scene's objects are located far from the camera plane and both foreground and background have insignificant differences in depth values. There are no artifacts due to the small shift of the objects in the scene.

## IV. CONCLUSIONS

We present a novel view synthesizing scheme which utilizes unique techniques, such as background decomposition in three layers, background edge dilation, vertical interpolation and edge aware warping, to improve the overall visual quality. Performance evaluations have shown that our method yields a significant visual improvement over VSRS for FTV.

## REFERENCES

[1] I. Koreshev, M. T. Pourazad, and P. Nasiopoulos, "Hybrid view-synthesizing approach for multiview applications," 3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), 2012, pp. 1-4.

[2] ISO/IEC JTC1/SC29/WG11 MPEG Document N11631, "Report on Experimental Framework for 3D Video Coding," Guangzhou, China, October 2010.

[3] M. Lang, A. Hornung, O. Wang, S. Poulakos, A. Smolic, and M. Gross, "Nonlinear disparity mapping for stereoscopic 3D," ACM SIGGRAPH, 2010, pp. 75.

[4] ISO/IEC JTC1/SC29/WG11, MPEG, "View Synthesis Software Manual," Sept. 2009, release 3.5.

[5] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," Computer vision and image understanding, 2008, pp. 346-359.

[6] http://www.stats.ox.ac.uk/~caron/code/bayesbt/ , 2017

[7] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: I. The method of paired comparisons," Biometrika, vol. 39, 1952, pp. 324-345.

[8] J. S. Lee, L. Goldmann, T. Ebrahimi, "A new analysis method for paired comparison and its application to 3D quality assessment," Proceedings of ACM Multimedia, 2011, pp. 1281-1284.

[9] V. Baroncini, M. Tanimoto, O. Stankiewicz, "Summary of the results of the Call for Evidence on Free-Viewpoint Television: Super-Multiview and Free Navigation," MPEG2016, Geneva, June 2016.

[10] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, No. 1, 1979, pp. 62-66.

 ISBN: 978-1-61208-533-3 