Smart Video Orchestration for Immersive Communication

Alaeddine Mihoub and Emmanuel Marilly
Alcatel-Lucent Bell Labs France, Multimedia Technologies Domain, Centre de Villarceaux, Route de Villejust, 91620, Nozay, France

Keywords: Camera Orchestration, Hidden Markov Model, Learning, User, Immersive Communication.

Abstract: In the context of immersive communication and in order to enrich attentional immersion in videoconferences for remote attendants, the problem of camera orchestration has been evoked. It consists of selecting and displaying the most relevant view or camera. HMMs have been chosen to model the different video events and video orchestration models. A specific algorithm taking as input high level observations and enabling non expert users to train the videoconferencing system has been developed.

1 INTRODUCTION

A key challenge of the telecommunication industry is to identify the future of communication. Immersive communication has been defined as the way to exploit video and multimedia technologies in order to create new relevant and valuable usages.

But in a context where the objective is to improve distant communications, sensorial immersion (i.e. all technical capabilities to mimic sensorial feelings) is not enough. Because communication is made of social interaction, narration, task driven activities, we need to include a new aspect for immersion: attentional immersion. Attentional immersion concerns the cognitive experience to be immersed in a narration, in a task or in a social interaction.

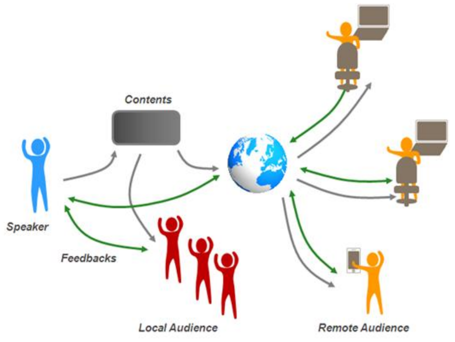


Figure 1: Remote Immersive meeting use case.

In order to improve sensorial and attentional immersion, the remote immersive meeting & experience sharing (e-education, town hall meeting) use case (Figure 1) has been observed and several pain points were identified such as keeping attention (e.g. interactivity, dynamicity, concentration, comprehention, boredom, diversion), remote audience feedback (e.g. reactions, questions, discussions) and video orchestration issues (e.g. how to switch between cameras?, which camera to displayed in the main view?, which metadata use?, How to model this metadata?).

In this paper we will focus mainly on the video orchestration issues.

2 VIDEO ORCHESTRATION

Having attentional immersion used for remote video presentation use cases (i.e. town hall meeting, e-learning, etc...) imply to develop and implement specific reasoning mechanisms. Such mechanisms enable for instance to identify which of the video events happening is the most relevant (Lavee, 2009) to display. Or, it may help to implement elements of the Cognitive Load Theory (Mayer, 2001) in order to support a better knowledge transfer (for instance when narration and visual information are complementary and presented simultaneously).

Our experimental video conference system has been extended to enable video orchestration supporting some of these attentional immersion aspects.

Several solutions and systems were proposed to solve the problem of camera selection/orchestration.

Mihoub A. and Marilly E..
Smart Video Orchestration for Immersive Communication.
DOI: 10.5220/0004230307520756
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 752-756
ISBN: 978-989-8565-47-1

For instance, a remote control has been chosen to select videos/cameras to display or pre-defined orchestration templates have been used to show participants of the meeting. Such exisiting systems are unable to manage high number of video streams with high level of details, dynamicity in the rendering, adaptability to the user intent and programmability and flexibility in the orchestration.

Video orchestration based on "audio events" is one way in this direction. Yet, as around 70% of all the meaning is derived from nonverbal behavior/communication (Engleberg, 2006) useful information for video orchestration are missing (i.e. gesture, expression, attention,…).

Al-Hames (Al-Hames and Dielmann, 2006) proved that the audio information is not sufficient and visual features are essential. Then, Al-Hames (Al-Hames and Hörnler, 2006) proposed a new approach based on rules applied on low level features such as global motions, skin blobs and acoustic features. HMMs (Hidden Markov Models) have been also used (Hörnler, 2009) for video orchestration by conbining low and high level features.

Based on theses observations and inspired from (Al-Hames and Hornler, 2007) and (Ding, 2006), we will use for our video orchestration a system based on HMMs taking as input only high level features such as Gesture (Fourati and Marilly, 2012), Motion (Cheung and Kamath, 2004), Face expression (Hromada et al., 2010), Audio (O'Gorman, 2010). The benefit of the use of high level features is to solve the problem of programmability of the video orchestration during video conferences. Basic users can define their own rules transparently and such approach improves the user experience, the immersion and efficiency of video-conferences.

3 PROGRAMMABILITY

Implicit or user intent-based programmability capabilities enabling to model video orchestration and to smartly orchestrate the displaying of video/multimedia streams have been implemented in our system. Data used by our HMM engine to model the video orchestration are captured through the combination of two approaches: visual programming and programming by example. In our HMM model, the transition matrix A contains transition probabilities between diverse camera views; the emission matrix B contains emission probabilities of each observation knowing the current state or screen; the initialization matrix π contains the probability for each camera to be showed the first.

3.1 Solution Description

Therefore, the "multimedia orchestrator" module, part of the videoconferencing system, has been augmented by the three following functionalities:

- Smart video orchestration capabilities thanks to HMMs.
- Learning/programmability capabilities. That means that the system is able to define automatically new orchestration models through user intent capture and interactions.
- Smart template detection. That means that the system is able to recognize the video orchestration model that best fits the video conference context/scenario and the user profile.

Figure 2 presents a basic scheme of the solution. The engine of the "Multimedia Orchestrator" module is based on specific mechanisms (e.g. learning mechanisms, scenario recognition,…) integrating HMMs.

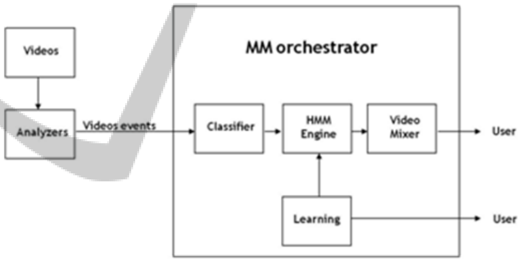


Figure 2: Basic scheme of the solution.

The "MM orchestrator" module takes as inputs video streams and video/audio events metadata (coming for instance form video/audio analyzers outputs). Video analyzers enable to detect high level video events such as gestures, postures, faces and audio analyzers enable to detect audio events such as who is speaking, keywords, silence and noise level.

Initially, based on the first received video and audio events metadata such as "speaker metadata", the classifier module selects the template that fits best the temporal sequence of events. By default, the user can select a model related to the current meeting scenario. During the use, the classifier can change the model if another one fits better the temporal sequence of events.

This problem of selecting the right model is known as recognition problem. Both, Forward algorithm (Huang et al., 1990) and Backward algorithm can solve this issue. In our MM orchestrator we have used the Forward algorithm. Next step after the selection of the best template is to select the most relevant camera to display. This decoding step is assured by the Viterbi algorithm (Viterbi, 1967). Once the decoding done, the HMM engine will orchestrate videos through a video mixer.

3.2 A New Learning Mechanism

In usual approaches (Al-Hames and Hornler, 2007); (Hörnler et al., 2009), the learning problem is known as an estimation problem. The EM algorithm (Dempster et al., 1977) (a.k.a. Baum Welch algorithm (Baum et al., 1970)) is used to reestimation the parameters (A, B, π). By default this process is done by experts and directly implemented in systems.

Figure 3 gives an overview of the proposed solution enabling basic users to create and personalize their own video orchestration models through the use of learning mecanisms (e.g. intent-based programming).

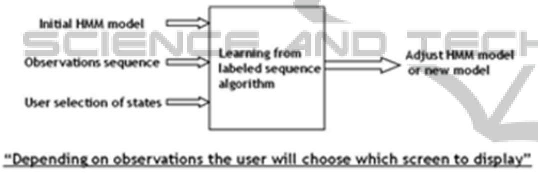

Initial HMM model
Observations sequence
User selection of states
Learning from labeled sequence algorithm
Adjust HMM model or new model
"Depending on observations the user will choose which screen to display"

Figure 3: Video Orchestration Learning Module.

A visual programming interface is providing to the user (figure 6). The interface displays the video streams and the detected video events. The user selects which video stream has to be displayed as main stream by the orchestrator depending of the detected video event. The learner module records the events and the corresponding chosen screens and generates a new template (or updates an existing one). From a technical point of view, the module records the observations and the corresponding selected states and generates a new HMM with the appropriate probabilities. The following section details the implemented learning process.

Learning Module Theory

The learning algorithm enables to create and train video orchestration models based on the user uses without any technical skills in progamming. It is composed of 3 modules: the user visual interface, the user activities recorder and the HMM generator. The three components (A, B, π) of the HMM has been determined in the following manner:

1. Training of the Initialization Matrix

The initialization probability of the first state selected by the user is set to 1 and the others to 0.

2. Training of the Transition Matrix

The training of this matrix is composed of 4 steps:

Step1: Get the number of states for the HMM inputted.

Step2: Generate a comparison matrix. This matrix will contain all possible transitions.

Step3: Browse the states sequence and each transition will be compared to each transition in the comparison matrix. If a similarity is found, the occurrence matrix will be filled.

Step4: Once the occurrence matrix obtained, the transition matrix is estimated. The equation 1 gives the formula enabling the transition matrix estimation.

$$a_{ij} = \frac{occ_{ij}}{\sum_{h=1}^{N} occ_{ih}} \tag{1}$$

Where Occ is the occurrence matrix.

3. Training of the Emission Matrix

For each state, each type of observation is count, and then divided by the total observations of that state. The equation 2 gives the formula enabling to estimate the emission matrix:

$$b_{ik} = \frac{occObs_{ik}}{\sum_{h=1}^{M} occObs_{ih}} \tag{2}$$

Where occObs represents the occurrence matrix for each type of observation knowing the state.

3.3 HMM Model for e-Learning

The Video Orchestration Learning module has been applied and tested in the context of a basic e-learning video conferences scenario. The scenario consists in one video stream for the lecturer/tutor, one video stream for the virtual class room and several individual video streams for the students/learners. Figure 4 gives a description.

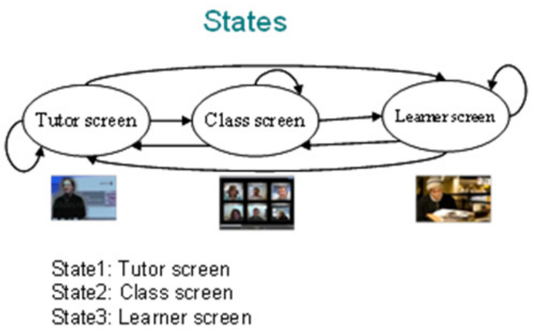


Figure 4: e-Learning use case description.

The HMM model is configured as follow:

- 3 States: 1-Tutor Screen, 2-Virtual Class Room Screen and 3-Learner Screen.
- 17 Observations. This number corresponds to the number of video or audio events that can be detected by our system. These observations are split in 7 families: Gestures, Motion, Face Expressions, Keywords, Audio Cues, Slide Number, Sub-Events. Figure 5 gives a detail representation of the observations used.

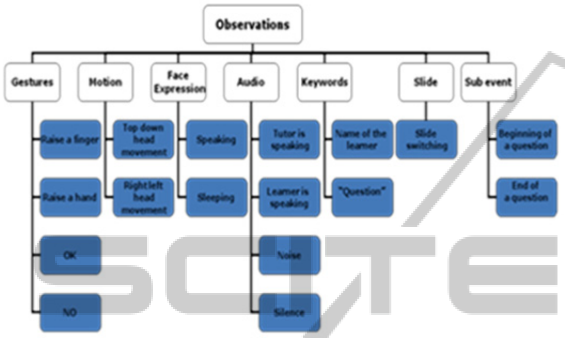
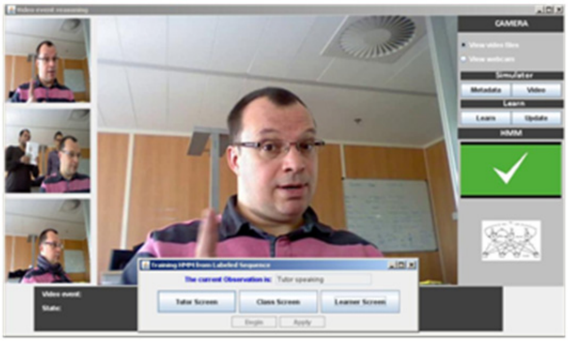


Figure 5: Model Observations.

For the scenario, 5 basics use-cases been defined corresponding to 5 initial video orchestration models which are: normal lecture, question/answer interactions, unsupervised question, exercise and learner presenting a work.

3.4 Evaluation

Figure 6 presents the graphical user interface of the learning module used to capture the user interactions and model the orchestration.



The current Observation is: Tutor speaking

Figure 6: GUI of the learning module.

Once the learning module implemented in the videoconference system, the performance of the HMM to correctly orchestrate the video streams has been evaluated. Table 1 gives an overview of the video orchestration performance per state.

The evaluation was based on K-Cross Validation (K=10). For 10 sequences, 24 observations for each one, we have in total 209 observations that have been well decoded and affected to the right state, so the global rate of a good detection is 0.87 (209/240).

Table 1: Evaluation of the Video Orchestration.

	Recall	Precision	F-measure
Confusion Matrix for Tutor State	0.97	0.86	0.91
Confusion Matrix for Class State	0.58	0.92	0.71
Confusion Matrix for Learner State	0.94	0.86	0.90

4 CONCLUSIONS

The paper highlights the interest of a learning module in the context of video orchestration with two main objectives: In the first hand enable user intent based programming to enhance the interactivity and the attentional immersion. In the other hand maintain good technical results. In addition to the learning module, the orchestration system was enhanced by a classification module enabling automatic detection of the appropriate scenario to make the orchestrator more flexible and more dynamic.

The next important step will consist in the usability evaluation. Qualitatively, the capability offers to the user to create or modify the video orchestration has to be evaluated in term of acceptance and interest. A lot of questions have to be considered, for instance: Did the user interact at ease with the module? Did he appreciate the use? Can we give to the user a total freedom in video orchestration? … A whole session for user testing will be organized in order to study usuability issues.

REFERENCES

Lavee G., Rivlin E., and Rudzsky M., "Understanding Video Events: A Survey of Methods for Automatic Interpretation of Semantic Occurrences in Video," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 39, no. 5, pp. 489 –504, Sep. 2009.*

Mayer, R. E., 2001, "Multimedia learning." *Cambridge University Press.*

Engleberg, I. N. and Wynn D. R., 2006, Working in Groups: *Communication Principles and Strategies.*

Al-Hames M., Dielmann A., Gatica-Perez D., Reiter S., Renals S., Rigoll G., and Zhang D., 2006, "Multimodal Integration for Meeting Group Action Segmentation and Recognition," *in Machine Learning for Multimodal Interaction, vol. 3869, Springer Berlin Heidelberg, 2006, pp. 52–63.*

Al-Hames M., Hörnler B., Scheuermann C., and Rigoll G., 2006, "Using Audio, Visual, and Lexical Features in a Multi-modal Virtual Meeting Director," *in Machine Learning for Multimodal Interaction, vol. 4299, S. Renals, S. Bengio, and J. G. Fiscus, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 63–74.*

Hörnler B., Arsic D., Schuller B., and Rigoll G., 2009, "Boosting multi-modal camera selection with semantic features," *in Proceedings of the 2009 IEEE international conference on Multimedia and Expo, Piscataway, NJ, USA, 2009, pp. 1298–1301.*

Al-Hames M., Hornler B., Muller R., Schenk J., and Rigoll G., 2007, "Automatic Multi-Modal Meeting Camera Selection for Video-Conferences and Meeting Browsers," *in Multimedia and Expo, 2007 IEEE International Conference on, 2007, pp. 2074 –2077.*

Ding Y. and Fan G., 2006, "Camera View-Based American Football Video Analysis," *in Multimedia, 2006. ISM'06. Eighth IEEE International Symposium on, 2006, pp. 317 –322.*

Fourati N., Marilly E., 2012, "Gestures for natural interaction with video", *Electronic Imaging 2012, Jan. 2012, Proceedings of SPIE Vol. 8305.*

Cheung S. and Kamath C., 2004, "Robust techniques for background subtraction in urban traffic video" *Electronic Imaging: Visual Communications and Image, San Jose, California, January 20-22 2004.*

Hromada D., Tijus C., Poitrenaud S., Nadel J., 2010, "Zygomatic Smile Detection: The Semi-Supervised Haar Training of a Fast and Frugal System" *in IEEE International Conference on Research, Innovation and Vision for the Future - RIVF , 2010.*

O'Gorman L., 2010, Latency in Speech Feature Analysis for Telepresence Event Coding" *in 20th International Conference on Pattern Recognition (ICPR), Aug. 2010.*

Huang X. D., Ariki Y., and Jack M. A., 1990, "Hidden Markov Model for Speech Recognition." *Edmgurgh Univ. Press, 1990.*

Viterbi A., "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *Information Theory, IEEE Transactions on, vol. 13, no. 2, pp. 260 –269, Apr. 1967.*

Dempster A. P., Laird N. M., and Rubin D. B., 1977, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal statistical Society, Series B, vol. 39, no. 1, pp. 1–38, 1977.*

Baum L. E., Petrie T., Soules G., and Weiss N., 1970, "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains," *The Annals of Mathematical Statistics, vol. 41, no. 1, pp. 164–171, Feb. 1970.*