In the previous chapter we have seen that the ability to search and browse content based on the social bonds is very important for making personal media more accessible. Nevertheless, it is too often the case that personal recordings are abandoned on memory cards or as downloaded files on hard drives never to be accessed again [19]. The main reason for this is that, as captured, video is not ready for being looked at. Video, as a time-based medium, necessarily requires processing after capture. Editing, for instance, can be performed on a handheld

---

1 This chapter contains extracts from the following papers:


smartphone to trim out poor and redundant content that is always captured alongside quality material. This is because video carries complex information and quality judgments cannot always be made on the spot, while filming. Editing is also required to create attractive artifacts for what we believe could later become valuable memories we want to watch and share with friends and family members. A simple juxtaposition of recorded fragments does not necessarily result in attractive mementos. But editing is not a simple process and people often do not want to engage with it, vide the results provided by our participants in Chapter 2. This is true for personal content from one source, but it is especially true when considering mixing content recorded at a single event from many sources: the community video problem.

This chapter provides an analysis of our efforts on multimedia authoring using community assets. As with browsing and navigation, we have developed a first version of an authoring system, subjected it to an extensive long-term user testing, and then developed an improved version that follows the guidelines of socially-aware multimedia authoring. As described in Chapter 2, our initial work was subjected to a 10-month evaluation process, enabling end-users to create stories reusing collective content for individual needs. Our initial results showed a general enthusiasm from participants, which were validated in the first evaluation phase. The initial implementation, which was aligned with the personal effort guideline, made use of a narrative engine to automatically compile personalized stories based on the community media assets [76]. While the video compilations produced by the initial system were considered visually compelling, end-users missed the capability of personalizing those by adding their own ‘imprint’. The complexity of authoring personalized stories from community assets have led to the consideration of the following research question:

**Question 1.4** Where is the balance between automatic and manual processes when authoring personalized narratives users care about?

We have approached this research question from three more concrete and strongly interlinked perspectives. In particular, this chapter investigates:

1. The degree to which media authoring can be simplified by the use of a narrative engine to produce a ‘rough cut’ (an initial video story) automatically;
2. The degree to which this rough cut can be automatically tailored based on the relationships within an end-user’s social network; and
3. The degree to which automatically generated video stories can be easily refined and further personalized using intuitive manual extensions with minimal extra effort.

The primary contribution of this chapter is a hybrid authoring system that allows users to create and share personalized media with others. This chapter is structured as follows. Section 4.1 motivates the problem of creating personalized stories from community assets, and discuss the evaluation we have carried out during the first phase. Section 4.2 describes the design and implementation of a new hybrid (or semi-automatic) authoring system that meets the functional user requirements elicited in phase 1. Section 4.3 reports on the results from the user evaluation of our prototype, demonstrating the benefits of our hybrid authoring approach. Finally, Section 4.4 concludes the chapter offering a discussion about the lessons learned.

4.1 Community-based Authoring

Creating compelling multimedia productions is a non-trivial problem. The problem is compounded when authors want to integrate community media assets: media fragments donated from a potentially wide and anonymous recording community. The purpose of this section is to describe our initial efforts to facilitate the creation of personalized stories from community assets.

Our initial approach provided users both independent manual and automatic authoring threads (called Editor and Composer, respectively). The intention was to compare the quality of easy-to-create fully automated compilations with the amount of effort required to manually creating personalized video stories.

Figure 4.1 shows Composer, the thread for automatically assembling video compilations in our initial prototype system. Users only had to explicitly select the subject matter (people, songs, instruments) and two other parameters (style and duration). Then, by pressing the ‘GO’ button, a narrative engine would be triggered, and in less than three minutes a video using the assets captured by different cameras at the concert would be created. The narrative engine would select the most appropriate fragments of videos from the repository, based on the declared user parameters, and assemble them following narrative constructs.
As mentioned in Chapter 2, the automatic authoring capabilities of the system were also assessed using expert input. Three video professionals with between 5 and 20 years experience were interviewed. All three agreed with the basic footage preparation and narrative structures that were used to build the video compilations. They were especially keen with the approach of using an audio track as a master timeline to drive the story development. They also concurred with our approach of automatically selecting alternative shots from cameras available in parallel tracks and using rules that selected clips based on shot types [76].

Our initial prototype system also provided an interface for manually creating video compilations (Editor). To find videos of interest, users could use the same set of filters and views available in the video exploration tool (refer to Chapter 3). Using the Editor, users could just drag and drop recommended video clips from the shared repository to the storyboard (see Figure 4.2). For example, a parent could add more clips in which his daughter was featured for sharing with grandma, or he could instead add a particularly ‘funny’ moment from the event when creating a version for his brother.
Figure 4.2. Initial prototype implementation for manually editing videos.

Figure 4.3. Elements of an authored video composition: the parent has included an introductory image and a video for making it more personal and intimate.
Apart from allowing fine-tuning of productions, the Editor also enabled users to perform enrichments. It provided mechanisms for including personal audio, video, and textual commentaries. For example, these could be subtitles aligned with the video clips commenting the event for others. Users could as well record an introductory audio or video, leading to more personalized stories. Figure 4.3 illustrates some elements of an authored video, where a parent has created his own version of a concert. He has also added some personal assets, such as an introductory image and a video recording of himself that acts as the message envelope. This functionality was called ‘capture me’.

As reported in Chapter 2, the evaluation of the initial system was preceded by 3 social events. While the first two recording experiments mainly focused on the evaluation of the annotation processes and narrative structures, the third one, a school concert in Amsterdam, allowed us to engage a group of parents, relatives and friends of performers for evaluating the initial version of our system. In the remaining of this section we discuss the lessons learned about the authoring threads during the evaluation of the first phase.

### 4.1.1 Phase 1 Evaluation

In this study, all participants first interacted with the community-based browsing interface (see Chapter 3), and then they were introduced to the authoring threads. In general, they appreciated both approaches to create personalized video compilations and considered the functionalities useful. Using our authoring tool they felt they could create more stories faster and easier (if compared to traditional systems – Q1.1-Q1.3 in Figure 4.4 from the evaluation of the first phase). Overall, the automatic assembled videos were considered visually compelling (see reactions in Figure 4.5). Although participants also indicated that they would like to have more manual processes available to further personalize and fine-tune the video compilations (Q1.4-Q1.6).

“I want more portraits of my daughter (in this automatic generated compilation)... is it possible to edit an existing movie (in the Editor)?”  
(Father of a performer)

In the manual authoring thread participants could find and select their favorite video clips. However, a complain was that they had to choose each and every clip for the compilation. Regarding optional processes, for most of our
participants the ‘capture me’ function – for including personal assets in a video compilation – was seen as a way to personalize videos for a target audience. As shown in the results (Q1.5), such functionality was mostly appreciated. Participants indicated they would use it, for instance, when creating a birthday present video.

In the initial version of our prototype system users could either generate video compilations automatically (not being able to change these later on) or edit manually (having total control but starting from scratch). While automatic compilations were quite appreciated because of shot selection and camera diversity, users provided important evidences that manual processes were indispensable to reflect intimacy and effort.

**Phase 1 Evaluation**

![Survey Results](image)

Figure 4.4. Results of the questionnaires from phase 1 evaluation.
Based on participants’ comments, reactions, and answers to the questionnaires, we can conclude that they appreciated the benefits of our authoring system and considered it a valuable vehicle for creating enjoyable memories. While these results were highly relevant, we were aware that they were not complete. More importantly was the indication that instead of the automatic or the manual authoring thread, a hybrid solution would better fit the participants’ needs (Q1.4). In the next section, we discuss the functional requirements that motivated the design of a new version of our socially-aware multimedia authoring system.

### 4.1.2 Requirements Gathering

Regarding manual authoring, participants identified a number of issues that could improve the creation process. Even though some participants were familiar with end-user video editing tools, for most of them this process was time consuming and complicated. Even though they appreciated the filtering functionalities included in the Editor, they indicated that they would not like to start the process from scratch.
Given the difficulties inherent in video editing, they would rather first use an automatic system that provided them with an already compiled story. Based on this feedback we introduce our first functional requirement:

i. *Not start from scratch*: users indicated their preference for an authoring paradigm, in which an initial narrative compilation would be created on their behalf. Such approach would simplify the authoring/editing task and increase their productivity;

Regarding automatic authoring, participants generally appreciated the easiness of use. The interface for automatically generating stories only required users to select a number of parameters such as duration, people, instruments, and songs to be shown in the compilation (see Figure 4.1). After a few minutes, users could watch a static narrative story based on their preferences. Even though they generally enjoyed the final results, they would have preferred that the system selected some of the parameters. In particular, they requested for automatic methods capable of identifying the interpersonal relationships with the performers of the concert. This discussion leads to our next requirement:

ii. *Consider implicit interpersonal relationships*: participants assumed that the system could automatically identify and process their interpersonal relationships with performers when creating video stories;

A common frustration with automatically generated videos in the initial prototype was that the automated process created a video story that could not be modified. Participants indicated that they would like to fine-tune (or personalize) automatic generated stories by using manual tools. They felt that the final result could potentially be more personal by adding assets and personal comments that more closely reflected their view of the event. This was of particular importance in video sharing situations, in which some participants wanted to send stories of the event to particular people within their social circle, such as an uncle or the grandmother of a performer. This result is consistent with our hypothesis that emotional intensity and intimacy should play a key role in socially-aware multimedia authoring systems (see Chapter 2). Geared by this discussion on personal effort we present our last requirement:
iii. **Allow for personal imprint:** participants suggested that automatically generated compilations could be modified. They wanted to remain in control over the final production, being able to make small changes. This approach would allow them to create more personalized stories.

Based on these requirements, we concluded that a new version of the authoring system was needed. The new approach would allow users to request a first compilation based on their implicit preferences and interpersonal relationships with performers. The system would then present an initial narrative, which could be edited and personalized on a per-clip basis. This hybrid authoring system ambitiously brings together both automatic and manual processes, so that narrative segments can be compiled, adjusted and edited successively. In the next section we discuss our efforts in designing and implementing such new authoring paradigm.

### 4.2 Hybrid Multimedia Authoring

The high-level workflow of our new authoring tool is detailed in Figure 4.6. Since we intend to improve the creation of video compilations based on multi-camera recordings, the input material still includes the school master track and the actual video clips that users agreed to upload. As shown in Chapter 3, all video clips are stored in a shared video repository that also serves as a media clip browser in which parents, students, and authorized family members can explore (and selectively annotate) the videos.

In the new design, the event exploration is the starting point for the authoring process. With the goal of creating a personalized video compilation based on a song, the user simply clicks on one of the songs in the concert program interface for triggering the narrative engine (Figure 4.7). The engine is in charge of creating a first montage from the video assets (and from video fragments) based on narrative structures and on interpersonal relationships (dependent on the identity of the user that is logged in). Such compilation, from now referred as the *Director’s Cut*, can be later modified by the end-user for making it more personal.

Next, we discuss how the three requirements identified over our initial prototype have been considered in the design and implementation of the new authoring system.
Figure 4.6. High-level workflow of our hybrid authoring tool.
4.2.1 Profiling Users

Profile of users logged in the system can facilitate the automatic creation of personalized video compilations. Traditional ways of user profiling include implicit activity monitoring (log) and explicit insertion of personal data. While these approaches provide relevant results for a statistically significant group of people interacting during a long time span, they are not sufficient for our highly personalized environment. For this reason, we have implemented a mechanism to automatically compute the relationships between users and performers.

Such mechanism follows three steps. First, we fill a database table with the songs each performer participated in. This is done by inspecting the annotations...
regarding performers in video clips, and looking for intersections with the songs that compose the event timeline. A key part of this procedure is that a weight is associated to each song/performer row in the database table. Such weight (or ranking) is calculated based on some parameters: the number of annotations each performer has in that particular song, the duration of these annotations (how long a musician is featured within a video clip), the quality of the annotated videos (e.g., high-definition or low-quality), and the shot type annotation (e.g., close-up or wide-shot). Note that the final ranking can be tweaked by giving different weights to the parameters. After all final weights have been calculated, they are normalized per song basis. This means that the performer with the highest weight in a song gets 1, while another that is not featured in the same song (or has the lowest weight) gets 0 (zero). All the other song/performer weights will then fall in the range [0, 1]. The result of this process is a table with normalized weights, which suggest the importance of each performer in each of the songs. The weighted song/performer table is used in the video selection process (compilation generation and alternative clip recommendations).

Second, we make use of the capturing behavior of each recorder individually. By taking into account the same parameters discussed in the first step, we model the behavior of a recorder towards the musicians in each of the songs. For that a similar database table, with an extra column (recorder) is used. By computing a normalized weight for each recorder towards each of the performers in each of the songs, we can derive their affection level, which as assumed, greatly influences the overall time a recorder spends capturing a specific musician. Based on these data, we can model relationships (was the performer his daughter? Was a friend of his daughter?), and thus provide information for the profiling process. Figure 4.8 shows the results of analyzing the metadata associated to the media captured during the high school concert in Amsterdam. In the figure, the recording behavior of a mother towards her kid is compared with the average behavior of the rest of the parents. We can observe that the affection level towards a performer is greatly influenced by the normalized weight of a particular recorder. In other words, the recording habits provide an important cue about the social relationships between recorders and featured performers.

Finally, the profiling process takes into account the user activity when browsing the shared media repository (e.g., videos a user watched, videos a user liked, most watched videos overall, most liked videos overall). This approach provides dynamic information when compared with the previous steps.
The information from these three steps is stored globally in the database and it is accessible by different engines. Based on normalized weights, inputs can be provided to the narrative engine, so automatic compilations do not only take into account narrative constructs, but as well interpersonal relationships between the users of the system and the people depicted in the video clips. This approach is directly aligned with our second requirement.

### 4.2.2 Automatic Generation of Stories

The first requirement identified in Section 4.1 was to provide automated authoring functionality, so the author does not have to start from scratch. Our system includes a reimplementation of the narrative engine used in the first phase. The new engine provides an initial story, as a playlist of video fragments. By itself, this functionality addresses our first functional requirement.

The narrative server wraps a narrative engine as a Web application, so that engine instances can be launched on the server. The Web application runs inside a generic Java Application Server (Tomcat) and it can handle request from other applications. These requests include the command dispatcher for starting/stopping the engine and the playlist dispatcher for requesting playlists. Further information
about the NSL language can be consulted elsewhere [51]. Figure 4.9 shows a video compilation created out of the ‘Adagio and Allegro from Sonata No 6 in E’ song.

As we will see below, the implementation of the narrative engine presented in Chapter 2 was modified to provide a set of alternatives (video clips) that can replace specific parts of the initial Director’s Cut, while still maintaining the narrative structure and the story line.

4.2.3 End-User Personalization of Stories

The third requirement we identified was the need for fine-tuning and further personalizing the automatically generated productions. To support manual personalization, the narrative engine does not only create a Director’s Cut, but it also provides a set of alternative clips that can potentially replace parts of the compilation (see Figure 4.10).

Once an initial compilation is ready, the user can modify it, allowing for personal imprint (third requirement). In order to enable such functionality we use a structured playlist format. In our work, we selected W3C’s SMIL playlist profile [17]. The benefit of SMIL is that it aims at integrating a set of independent multimedia objects (in our case video fragments) into a synchronized multimedia
presentation. It contains references to the media items, not the media content itself, and instructions on how those media items should be combined spatially and temporally. Other approaches on video mashups typically provide a final encoded video item, in which it is not possible to modify or enrich individual sequences. In our case, the richness of the SMIL language permits the user to perform dynamic operations on the initial video stories by simply modifying a text document (the SMIL file). The actual process of manipulating the document is hidden from the author, who simply sees an interactive user interface in the browser’s Web page.

The video compilation generated by the narrative engine contains a set of references to video fragments (using clipBegin and clipEnd parameters). In addition, it provides a number of switch containers (<switch>) that contain the alternative clips (or set of clips), which can be selected for personalizing the initial story. Such alternative video clips have been selected by the narrative engine, so the narrative intent is not lost. For example, it will offer the option of selecting a different camera angle or of selecting a different point/person of interest. In addition to these features offered by the narrative engine, the end-user can decide to perform more radical modifications by adding other assets from the database or
by enriching the video compilation (e.g., adding comments). All these modifications will be incorporated into the original SMIL file. For viewing purposes we use the Ambulant Player, which provides a full implementation of the SMIL language (see Figure 4.11). The benefit of using SMIL is that the recipient of the video can easily further enrich and modify the video compilation, and send it to others or maybe return it to the original author, enabling reciprocity. In Chapter 5 we will discuss our efforts to support personalized end-user enrichment within third-party content.

The combination of a profiling infrastructure based on interpersonal relationships, a narrative engine capable of creating attractive video compilations, and the use of manual mechanisms for tweaking and personalizing such compilations results in a unique authoring tool. The validation of this authoring tool for creation of highly personalized (but compelling) productions characterizes the major contribution of this chapter, as reported in the next section.

Figure 4.11. Director’s Cut: song player.
4.3 Evaluation

Nine (9) participants, enrolled in the second phase of the evaluations, filled in the questionnaires about the Director’s Cut functionality (for more information about the evaluation process, please refer to Chapter 2). Based on our observations, responses to the questionnaires, and analysis of the collected audio/video material from the interviews, in this section we present the results and discuss the findings from the evaluation process.

4.3.1 Results and Findings

Figure 4.12 shows the answers given by the participants after making use of the Director’s Cut functionality. In general, all participants appreciated the new prototype (Q2.1). Six participants said that the Director’s Cut offers a better way to edit videos if compared to existing video editing software they know (Q2.2). The other 3 users claimed they were unfamiliar with such tools, and therefore, they were unable to judge.

Again, similarly to the results obtained with the initial system (Q1.1-Q1.3), almost all participants argued that they would create more video stories (Q2.3) and quicker (Q2.4) because the tool was easy to use (Q2.5).

“It was very easy to use and it selected which videos I wanted well.”
(Brother of a performer about the automatic generation component)

“Very easy to use (editing based on alternative clips). I wouldn’t want to spend hours looking at a help menu. This was simple enough for me.” (Mother of a performer)

When asked whether they would add themselves to personalize a story (Q2.6), 6 users, mainly youngsters, mentioned this would be a good functionality. However, our senior participants claimed they would not do so. A similar feedback was obtained in the first evaluation process (Q1.6).

“It would be interesting to have a functionality to add other videos (that not only the ones suggested).” (Father of a performer)
Figure 4.12. Results of the questionnaires in Woodbridge (UK).
All participants indicated they would like to create different types of productions (Q2.7). When questioned about the types of video stories they envisaged, the ‘song-based’ video came up as first choice among most of them. Some argued that depending on the social situation they would create and share different versions with different audiences.

“If my family misses my performance I would send the full performance to them… but if I want to send it to my singing teacher I would share a more focused version.” (Performer)

Figure 4.13 shows some participants during the evaluation process. A one-to-one comparison between the first and the second phases would be unfair (different users, events, tools). What we can say is that in the second evaluation both the automatic generation of initial video stories and the manual tools for tweaking had extremely good scores (Q2.8 and Q2.9 respectively). These results provide strong evidences that a hybrid framework builds on the best of each approach: assisting on complex tasks (start from scratch) but still making sure the user plays an active role in the process whenever desired (personal imprint).
4.4 Discussion

Creating compelling multimedia presentations remains a complex task. This is true for both professional and personal content. For professional content, extensive production support is typically available during creation. Content assets are well structured, content fragments are professionally produced with high quality, and production assets are often highly annotated (within the scope of the production model). For personal content, nearly none of these conditions exist: content is a collection of assets that are structured only by linear recording time, of mediocre technical quality (on an absolute scale), and with only basic automatic annotations.

The problem is made worse when authors use community assets of an event. In events such as high school concerts, a single concert can generate hundreds of video clips, taken from multiple vantage points, using tens of cameras. With our initial prototype we could generate syntactically correct automated stories that served generic needs (much like a conventional video mashup). Our users found these compilations compelling but not their own: they missed a personal touch.

In this chapter, we reported on a hybrid authoring approach that provides mixed support for automated creation (requirement i) and manual enhancement of personalized video stories (requirement iii). We targeted small-scale events, where lightly annotated assets are provided. Our assumption is that editors at these events will want to highlight personal aspects: a particular instrument, a particular child, a particular solo (or goal). This places demand on a system to help users to select appropriate content of personal interest (requirement ii), and to help build compelling stories with minimum effort (in accordance with the personal effort guideline presented in Chapter 2).

We acknowledge there are some limitations regarding the amount of automated personalization that a system can provide. Abstractly, given unlimited personalized annotations and unlimited information on all members of a potential target user community, we suspect that great strides could be made in automated personalization. The reality is, however, that for community assets, personalized annotations are limited, and the target user group is lightly profiled. This requires an interface that allows direct user intervention in creating content.

Providing direct user intervention has tremendous benefits: the user best knows his/her target audience. The differences between uncle Henry’s interest and those of Grandma are often clear in the head of the human author, but largely inaccessible to an automated system. At the same time, end-users have only a limited amount of time and energy to create personalized stories (many are busy
recording new content, rather than editing old content!). This requires a balance of complexity and functionality. We feel that our approach provides this balance. Based on user feedback as part of our four-year study, we feel that we have shown that it is possible to satisfy casual content creators while still allow extensive personalization to take place if needed. These results directly answer our research question (and fulfills the requirements on effortless interaction, personal effort and intimacy introduced in Chapter 2). We feel that the combination of automatic and manual processes is unique and powerful.

While concentrating in the creation process, we cannot forget that multimedia sharing can also stimulate user comments and reactions, which is as well part of the authoring workflow. This is the topic of next chapter, in which we present our efforts on empowering users in commenting within personalized multimedia presentations.