Secure, Portable, and Customizable Video Lectures for E-learning on the Move

Marco Furini
Department of Social, Cognitive and Quantitative Sciences
University of Modena and Reggio Emilia
Via Allegri 9 - 42100 Reggio Emilia, Italy
E-mail: marco.furini@unimore.it

Keywords: e-learning, secure video lectures, media in education

Received: September 27, 2008

The production of video lectures for the mobile scenario is becoming popular, as the pervasiveness of mobile technologies is making learning independent of time and space. In such a scenario several challenges need to be addressed, and the contribution of this paper is MOLE, an architecture that produces secure, portable, and customizable video lectures for generic mobile devices. Video lectures are produced with a format that ensures play out compatibility in most mobile devices, and with a security mechanism that protects contents from un-authorized usage. Furthermore, a video lecture is organized in such a way that a learner can adapt the lesson development to his/her learning needs by locally interacting with the system, which means that people with different needs may use the same video lecture file. A prototype implementation of MOLE shows its feasibility. The production of secure, portable, and customizable video lectures may help expanding mobile learning as generic mobile devices may be used as learning tools.

Povzetek: Video predavanja za e-učenje.

1 Introduction

Mobile learning, the combination of mobile computing and e-learning, is expected to expand and to evolve dramatically over the next few years. Exploiting the pervasiveness of mobile technologies it is possible to make learning independent of time and space so as to make mobile learning a real opportunity. As a result, the production of video lectures is becoming more and more important, as video is one of the most powerful media to present information and students find video materials very compelling [1].

To make mobile learning a success, several challenges are still open and need to be addressed: mobile learning may favor technologically advanced students; the large variety of learning devices my cause lectures to be encoded in several formats; being a digital media, video lectures might be subject of intellectual properties and copyright issues [2,3].

In this paper we propose MOLE (MObile LEarning), an architecture to produce *secure, portable*, and *customizable* video lectures for the mobile environment. MOLE aims at producing video lectures for generic mobile devices, i.e., devices with different computational and storage characteristics, from simple video players to PDAs, from iPods to smart phones. Security is a key feature in nowadays mobile scenario, as video lectures may contain copyrighted materials. Therefore it is necessary to protect such material from un-authorized usage. Since the mobile scenario is filled with devices that have limited computational resources (e.g., cellphones), the usage of complex security mechanisms may be a burden for most devices. For this reason, MOLE is designed with a security mechanism that is light enough to be used over generic mobile devices. Portability is another important feature, as people are equipped with a large variety of devices. In such a scenario it is not reasonable to produce several versions of the same video lecture so as to meet the characteristics of students' devices. MOLE aims at producing video lectures with a format that can be played over the majority of mobile devices. Similarly, in addition to the heterogeneousness of mobile devices, the mobile scenario is filled with students who have different learning needs. Also in this case, it is not reasonable to produce, for the same subject, several video lectures with different learning levels. However, content adaptation is very important in learning as it allows a better supporting of learners with different skills and motivations [4,5], and, if not provided, remote students would feel frustrated and would tend to drop courses [6,7]. For this reason, MOLE organizes the contents of a video lecture in such a way that it contains several learning levels, giving the student the opportunity to tailor the lesson to his/her learning needs.

The feasibility of MOLE is tested through a developed video lecture player. Results show that MOLE produces video lectures for the mobile scenario with

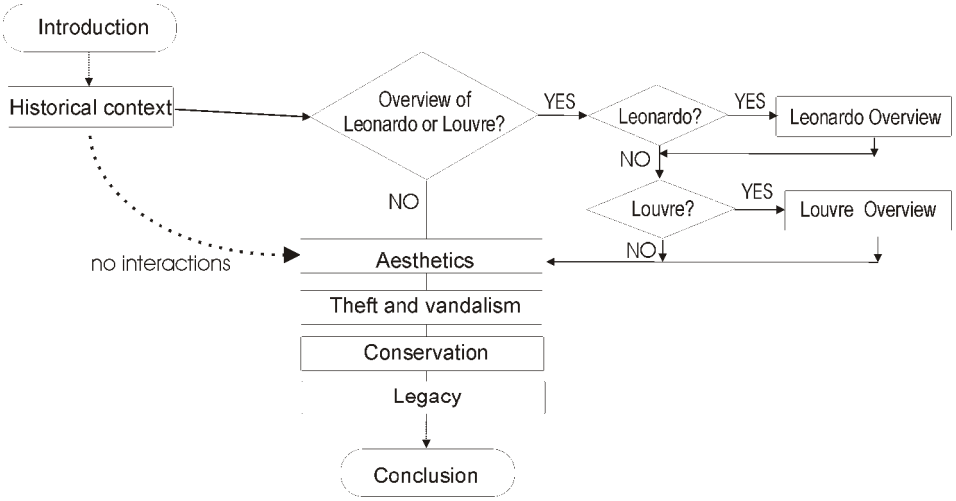

Figure 1: Classroom interaction during a lesson related to Leonardo's artwork The Mona Lisa. The lesson can take different directions, according to students' request. If interactions are not provided, the lesson has only one direction.

characteristics that potentially transform any mobile device into a learning tool.

The remainder of this paper is organized as follows. In Section 2 we briefly overview proposals in the field of mobile learning; Section 3 presents details of the MOLE architecture, whereas its feasibility investigation is discussed in Section 4. Conclusions are drawn in Section 5.

2 Related work

The effects of mobile technologies in learning have been investigated under different prospectives [6]: new models for teaching and learning (e.g., [8,9]); effects on the design process and on student experience (e.g., [10,11]); effectiveness and costs of using mobile devices in education (e.g., [12]); new tools for distance learning (e.g., [13,14,15,16]); adaptation of learning contents to mobile devices (e.g., [17,18,19]). Since our approach deals with portability, security, and content adaptation, in the following we focus on approaches that investigate the same issues.

Marinelli and *Stevens* [20] focus on customization; they propose segmenting a video lesson into small chunks, which are stored on a video server. When a student access to the video lesson, depending on his/her choices, the most appropriate video chunks are streamed. This approach requires a data transfer rate able to stream the video content. Unfortunately, while effective for wired environments, the streaming of a video lecture may be problematic in the mobile scenario: data transfer rate may be insufficient (e.g., in rural area, where high speed wireless networks are not still available), not to mention that data traffic is usually metered and paid by megabytes. Furthermore, this approach requires a play out device with communication facilities, and thus it cuts out a considerable number of mobile devices (e.g., Ipods). *Liu* and *Choudary* [21] focus on the data transfer problem and propose propose encoding video lectures with different quality levels and with different compression ratios, so as to meet different transfer data rate and different mobile devices. Although ameliorating the problem of streaming video lectures in a mobile scenario, this approach requires a device with communication facilities. *Kung* and *Wu* [22] focus on customization too, and they propose integrating synchronous and asynchronous learning systems to provide content adaptation to student's needs. *Weippl* [23] focuses on security and analyzes the weaknesses inherent to mobile devices.

The novelty of MOLE is that streaming is not used, mobile devices do not need communication facilities, and security is guaranteed with a light security mechanism.

3 The MOLE architecture

In this section we present details of MOLE (MObile LEarning), the architecture designed to produce *secure, portable*, and *customizable* video lectures for the mobile scenario.

Before presenting details of the architecture, let us consider a simple lesson which we'll refer to in the remainder of this paper. The lesson is about *Leonardo's artwork The Mona Lisa*. Briefly, it involves different topics: historical context, aesthetics, theft and vandalism, conservation, and legacy. Since the part related to the historical context contains topics explained to students (e.g., Leonardo history, Louvre museum) in previous lessons, a classroom lesson usually develops according to whether the students have clear these topics or not. If not, the teacher usually spends some minutes in reviewing them, without entering into the (already explained) details. Hence, the classroom lesson may develop differently, as depicted in Figure 1: a pre-defined lesson path goes from *historical context* to *aesthetics*, but due to students/teacher interactions, other paths are possible. These additional paths represents different learning levels that may be used by students to adapt the video lecture to their learning needs.

In most of distance learning systems, if a student has not clear a subject explained in previous lessons, he/she has to stop playing out the current video lecture, has to

browse the library to find out the right video lecture, has to browse the video to find the points of interest and, finally, has to re-watch it. Needless to say, it is likely that most students will continue watching the video lecture even though they don't clearly have the picture of a particular subject. Due to the importance of content adaptation MOLE allows simulating the classroom scenario also when students and teacher are distributed in time and space.

The idea is to store several possible lesson paths into a single video lecture file and on defining points where a student can modify the lesson development by interacting with the system. This means that all the different lesson paths are stored in a single file.

Figure 2 shows the architecture of MOLE: a video lecture is produced by first recording/encoding the educational material; then the material is organized through a text-based script; finally, the material is stored in a multimedia container and protected against un-authorized usage. At the student-side, once the video lecture is downloaded, the first step is the removal of the protection, so as to allow the retrieval of all the contents (educational material and content organization). Then, the video lecture is played out, and according to the student's interactions, the content of the video lecture is adapted to the student's learning needs.

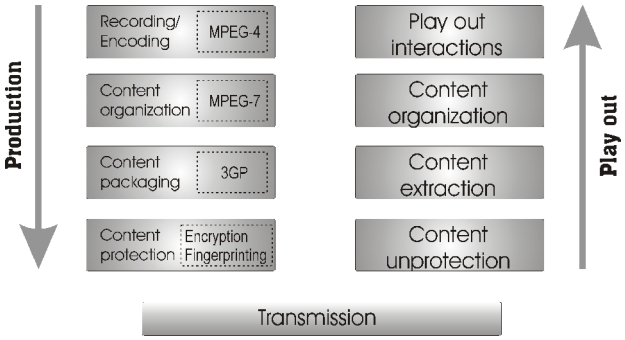

Figure 2: The MOLE architecture to produce and protect video lectures. To guarantee large compatibility in most mobile devices, video lectures are produced with well-know standards mechanisms (dotted box).

3.1 Production of the video lecture

3.1.1 Encoding of the audio/video material

To reduce both the download time and the storage space, a video lecture should have low bit rate, while providing a good video quality. To this aim, MOLE uses MPEG-4, a standard designed to encode audio/video contents with high quality at low bitrates [24]. Since a detailed description of this standard goes beyond the scope of this paper, here, we simply highlight that MOLE encodes video with MPEG-4 Part 2 specifications, and audio with MPEG-4 Part 3 specifications. Thanks to the exceptional performance and quality of Part 2 and Part 3, and to the large presence of MPEG-4 players in mobile devices, MOLE produces video lectures that are limited in size and playable over most modern mobile devices.

3.1.2 Organization of educational material

Since there is no way to know in advance the different learning needs of the people who will watch the video lecture, it is necessary to include, inside the video lecture, points where students can virtually interact with the teacher (and actually interact with the system) so as to provide different learning paths. The organization of the educational material is done through a text-based description which is called *lesson script* and is produced according to the teacher's experience.

The description virtually divides a lesson into several video chapters. Similarly to the IEEE's learning object model, a video chapter contains a portion of the lesson that can be used once or several times during the lesson play out, depending on the defined script and on the student requests. However, it is worth noting that all the video chapters are part of a single video file and hence there is no need for the student to get multiple files or to be on-line to receive the most appropriate video stream. In fact, it is the player that uses the lesson script to jump from one video chapter to the other depending on the student's interaction.

MOLE defines four different types of video chapter (Figure 3), with the following meaning.

- **Initial**. It's the chapter that begins the lesson; it has only one possible outgoing lesson direction and only one initial chapter per lesson is allowed;
- **Interactive**. It's the chapter that allows a student to virtually interact with the teacher in order to modify the lesson development. It has multiple possible incoming and outgoing lesson directions. Several interactive chapters per lesson are allowed;
- **Sequential**. It's the chapter that simply plays a portion of the lesson. It has multiple possible incoming lesson directions, but only one

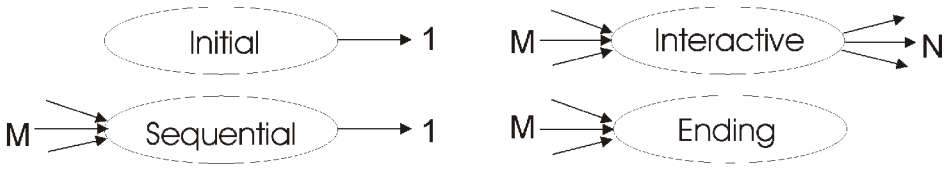

Figure 3: Types of video chapter that may compose a video lecture. The number of possible incoming and outgoing lesson directions characterize the typology.

possible outgoing direction. Several sequential chapters per lesson are allowed;

- **Ending**. It's the chapter that ends the lesson. It has multiple possible incoming lesson directions. Only one ending chapter per lesson is allowed.

Thanks to interactive video chapters, a student can modify the lesson development according to his/her learning needs. For instance, Figure 4 presents three possible lesson developments: student \#1 only requires an overview of *Leonardo*, student \#2 does not require any material overview, whereas student \#3 requires *Louvre* overview. This simple example shows that the organization of the contents allows having different lesson path inside the same video lecture.

Once the material has been divided into several chapters, the actual lesson description takes place and specifies: i) all the information that describe a video chapter (e.g., beginning and duration time, possible textual information associated, etc.), and ii) the points where a student can modify the lesson development.

Different description languages may be used to produce the actual lesson description (e.g., XML, SCORM, MPEG-7), all of them with pros and cons. For the sake of clarity, in the following we present examples described based on the MPEG7-MDS standard [25]. This standard is composed of metadata structures and is used to produce a description of the spatial layout of different media objects (e.g., audio, video) and of the temporal order in which these objects will be played out.

```
<VideoSegment>
  <label>"Z"</label>
  <RelatedMaterial>
    <MediaLocator><MediaUri>
      http://.../mm/monalisa.pdf
     </MediaUri></MediaLocator>
   </RelatedMaterial>

   <MediaTime>
      <MediaTimePoint>
        00:22:30
      </MediaTimePoint>
      <MediaDuration>
        00:03:30
      </MediaDuration>
   </MediaTime>
   <TextAnnotation>
      <FreeTextAnnotation>
         The Mona Lisa painting
         can be found at page 356.
      </FreeTextAnnotation>
   </TextAnnotation>
</VideoSegment>
```

Table 1: The usage of MPEG7-MDS to describe a portion of a video lesson.

The description is text-based and uses tags (in the form of `<tag [attribute=value]>`) to define properties of a media object (or of a part of it). For instance, Table 1 shows a video chapter description related to *Mona Lisa*: it begins after 22 minutes and 30 seconds and last 3 minutes and 30 seconds. Additional information, like a text description `textannotation` tags) or related material (`RelatedMaterial` tag), can be attached to any chapter.

To define the set of possible points where a student can modify the lesson development, MOLE uses a table called *scene transition table*. Each entry of this table is uniquely identified with a Video Chapter Identifier (VCI) number and includes the possible question asked by the teacher to begin the interaction with students and a set of possible destinations the lesson can take based on the student's answer. For instance, the entry related to video chapter *Y* (Fig. 4) has two possible destinations (chapters *Y.1* or *Z*), whereas the video chapter *X* only has a single chapter destination (chapter *Y*). In this way, a student can get get a lesson tailored to his/her needs, by simply selecting the most appropriate video chapter for his/her learning process. Note that also the scene transition table is described with textual information.

3.1.3 Protection of the video lecture file

To protect contents from illegal usage or from un-authorized modifications, MOLE is equipped with a license-based security mechanism (a.k.a. Digital Right Management, DRM for short) that discloses interactive materials only to authorized students.

The basic idea of the security mechanism is to wrap a media file with encryption, code authentication and

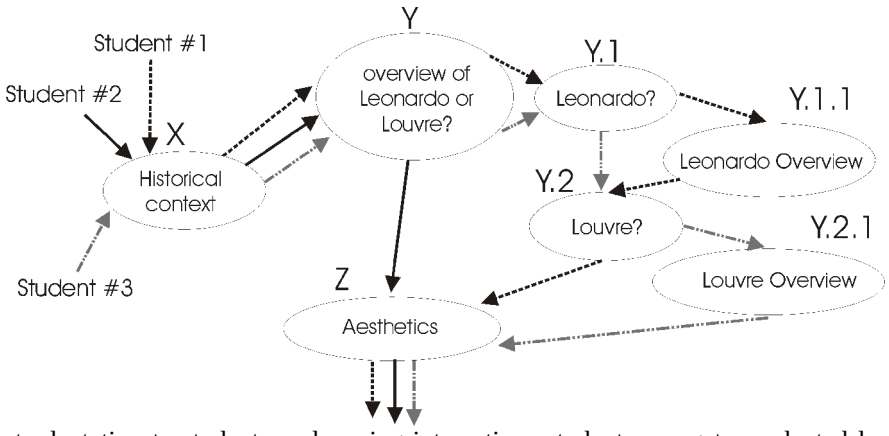

Figure 4: Content adaptation to student needs: using interactions students can get an adapted lesson. Here, three different lesson developments are shown.

information hiding, as depicted in Figure 5. MOLE gives the choice to provide the single pre-defined lesson path in clear so that every ordinary player can access to and can play out the single pre-defined lesson path. This feature may be useful to those organizations that want to give students the opportunity to appreciate the educational material so as to tempt them to buy the right to play out all the lesson paths. However, it is worth noting that the single pre-defined lesson path in clear is an option and is not mandatory (the single pre-defined lesson path may be encrypted as well as all the other lesson paths). In particular, MOLE organizes the video material as follows:

- All the video chapters that compose the single pre-defined lesson path are stored in the first part of the video file. These chapters might be in clear (so that every ordinary player can play them out) or might be encrypted with a symmetric technique;
- All the other chapters are encrypted with a symmetric technique and are stored in the second part of the file;
- The first and the second part of the video file are separated by 60 seconds of blank video, so that if the pre-defined lesson is in clear, an ordinary player will not produce an immediate play out error when trying to play out the encrypted second part of the video file.

The encryption/decryption key is hidden inside the first part of the video, so that only players able to retrieve the key can play out the video lecture;

Before presenting details of how the second part of the video file is encrypted, for the sake of clarity, let us review how a license-based mechanism usually works: first, the content provider generates a symmetric key and then encrypts the media content (the second part of the audio stream and the the lesson script in our case). After the encryption, the symmetric key is hidden inside the media file and a license file is generated with all the information needed by the decoder to play out the media file (users rights, positions of the media file where to find the hidden key, etc.). Finally, the license file is encrypted with an asymmetric key technique so as to bind the license file to the owner of the private key. License acquisition can be done in a transparent way, as it happens today with several DRMs (e.g., Microsoft DRM 10).

Figure 5 shows details of the MOLE security mechanism. The key used to encrypt the second part of the lesson file and the lesson description (α in our case) is hidden into the first part of the media file using a watermarking technique that spreads the hidden key without compromising the media content (e.g., [26]). To spread the information, this technique generates the so-called *watermarking key* (WKey) and uses it to hide the information. This key is stored inside the license file, along with information used to check data integrity. To guarantee data integrity the content provider uses a hash function H to compute the hash value of the encrypted video stream (i.e., $HV=H(E_\alpha(video\_stream))$) and of the lesson script (i.e., $HD=H(E_\alpha(lesson\_script))$) and stores them inside the license file. These values are also watermarked inside the media file and inside the lesson description, as they will be used by the video lecture player to check the integrity of the stream and of the lesson script.

Finally, the license file is encrypted with a public/private key, so that only who owns the private key can decrypt it. We recall here that the private key is usually given during the sign-up procedure and is stored in the device repository key, which is hidden with suitable software engineering techniques.

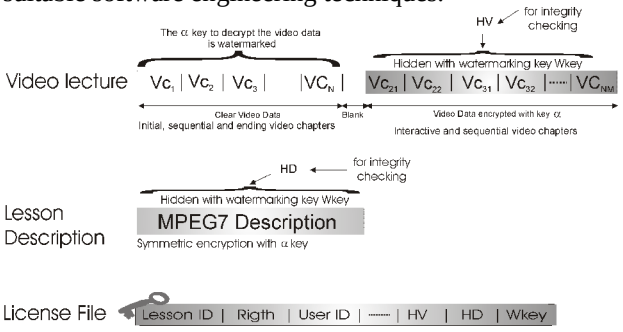

Figure 5: Security mechanism: the first part of the lesson file might be in clear or encrypted, whereas the second part of the lesson file is encrypted. The decryption key is hidden in the first part of the file. The lesson description is encrypted with the same key used to encrypt the second part of the lesson file. The license file is encrypted with private/public key and contains information to decrypt the lesson file.

3.1.4 Packaging of educational contents

Video material and lesson description need to be stored in a single multimedia container. MOLE uses 3GP, a multimedia container designed to handle multimedia contents in a mobile environment [27]. It has been defined by the Third Generation Partnership Project (3GPP), which provides worldwide standard specifications for multimedia contents over 3rd generation cellular networks. A 3GP file can contain material encoded with different schemes like H.263, MPEG4, MP3, and is supported by the majority of multimedia mobile devices. The file structure is composed of data structures called boxes, which are hierarchically organized. Each box is identified with a tag and contains a media object (e.g., audio, video), which can be actual media data or simple metadata (information to describe the media properties). MOLE stores the video material in a `trak` box and the lesson description in a `udta` box. For details about the 3GP file structure we refer the readers to [27].

3.2 Play out of the lesson content

To enjoy the full features of the lesson file, an enhanced player is necessary. As shown in Figure 2, the player is in charge of removing the protection so as to access to

educational material. In particular, the player is in charge of: i) checking the integrity of the video file, ii) decrypting and playing out the lesson file, iii) interacting with the student and jumping from one chapter to another depending on the student's choices.

To check the integrity of the video file as well as of the lesson script, the player first uses the student's private key (note that private keys are usually stored into the device repository) to decrypt the license file. Once decrypted, the license file provides, among other information, the watermarking key and the values to check the integrity. At this point, the player retrieves *HV* and *HD* from the lesson file and the lesson description and compares them against the values retrieved from the license file. If the integrity check fails, the play out is interrupted, otherwise the player retrieves the hidden α key and begins the lesson play out.

While playing out the video lecture, the player retrieves the video chapter information from the MPEG-7 description (i.e., chapter label, media time, etc.) and from the scene transition table (i.e., question and destinations, if any). If the chapter is interactive, depending on the option selected by the student, the player jumps to the video chapter in order to continue with the lesson.

4 MOLE prototype implementation

In this section we present a prototype implementation of MOLE and a security evaluation.

4.1 Video lecture production

To evaluate MOLE we produce a video lecture encoded with MPEG-4 at 15 frames per second, with a resolution of 176x144 pixels (common display resolution in most mobile devices) and a bitrate of 64 kbps; audio is MPEG-4 encoded (Part 3, commonly known as AAC-LC), two channels at 128 kbps. The overall bitrate of the video lecture is of 192 kbps. Content organization is specified with MPEG7-MDS, and content protection is achieved with the MOLE security mechanism. All the contents are then stored on a 3GP multimedia container.

4.2 Player implementation

Using the Nokia Prototype SDK 4.0 for Java ME, we develop a player able to play out video lectures produced with MOLE. Figure 6(a) shows the play out of a non interactive chapter, whereas Figure 6(b) displays that the play out of an interactive chapter changes the menu option. When the *ask* button is pressed, the player presents a multiple choices menu (Figure 6(c)), where a student can select his/her preferred subject. Once selected, the player jumps to the associated video chapter.

(a) (b) (c)

Figure 6: Lesson play out: (a) sequential chapter; (b) interactivity activates the *ask* button; (c) selection of a particular topic.

4.3 Feasibility Analysis

A first investigation analyzes whether the security mechanism can be sustained by current mobile devices. In fact, security operations like decryption use complex mathematical operations and can be problematic to mobile devices with limited processing power. For this reason, we analyze the decryption processing cost in relation to recent released PDAs (processors speed that ranges between 126 and 624 MHz) and cellphones (processors speed around 200 MHz)[1], in order to investigate the feasibility of our approach.

Video stream and lesson description are encrypted with symmetric technique. *Ravi* et *al*. [28] showed that a 206MHz SA-1110 processor can sustain a decryption rate of 1.8 Mbps per second, when fully dedicated to the task, and a decryption rate of 180 kbps when only 10% of the computational resource is dedicated. Hence, the decryption workload of the video stream (192 kbps) can be sustained by new generation PDAs and the cellular phones.

In order to handle interactions, the lesson script has to be decrypted during the play out of the first two chapters (in fact, since the first chapter is not interactive, the play out of the first two chapters is guaranteed). Since the description of a single video chapter is very short, it is very likely that the decryption workload of the lesson script can be sustained. For instance, 100 video chapter descriptions (as the one reported in Table 1) can be decrypted in just 2 seconds (considering a decryption rate of 180kbps). Since it is reasonable to assume that the first two video chapters last much more that two seconds, and that a lesson does not have thousands of video chapters, it is safe to assume that the decryption workload of the lesson script can be sustained by recent PDA and cellphones.

The license file is encrypted with asymmetric technique, which uses more intensive mathematical operations than a symmetric technique. A study performed in [29] showed that an asymmetric decryption takes 2.63 ms for 1 KB of data, using a 100 MHz processor. Again, since this file is usually very small (around 2-4 KB), portable devices can sustain the decrypting of the license file without causing an excessive delay to the content material play out.

A second investigation analyzes the storage space required by the proposed approach. In fact, before beginning the play out, the video lecture has to be

[1] For instance, the Nokia N90 serie is equipped with a 220 MHz processor, whereas the iPhone has a 620 MHz processor as described at: http://www.engadget.com/2007/07/01/iphone-processor-found-620mhz-arm/

entirely downloaded, and the storage space required might be problematic to mobile devices. Since the overall bitrate of the video lecture is of 192 kbps, a one-hour video lecture requires around 86 MBytes. Even though a video lecture file may contain several lesson paths, it is reasonable to assume that a video lecture file requires space on the order of hundreds of MBytes. Recent released PDAs and cellphones are equipped with an internal memory storage that is on the order of GBytes[2], or are equipped with a slot where users can insert SD memory card (which are becoming very popular for their limited cost). Therefore, it is safe to assume that the space storage is not a burden for the proposed approach.

A final investigation analyzes whether the security goal is achieved and under which conditions. By assuming the existence of cryptographically secure hash functions and of a secure symmetric/asymmetric key encryption scheme, sharing of a video lecture and/or a license file is useless. In fact, one can successfully share it if he/she can capture both the video and the script description, but to break the protection, the adversary must learn the watermarking key of the destination player. As previously mentioned, this is hard with no knowledge of the the watermarking key. Furthermore, also alteration of the content rests on the security offered by the watermarking scheme. In fact, alterations are possible only if new integrity parameters (i.e., the hash value of the video lecture and of the lesson script) can be stored.

It is worth mentioning that security of digital material is a problem that admits no final and provably strong solution. Briefly stated, the reason is that the security mechanism works in an untrusted environment (the playout device is under the user's control). Readers can refer to [30] for a broad and interesting survey paper on effectiveness of security mechanisms. It also worth mentioning that watermarking techniques are also subjects of a never ending debate: The biggest threat faces the removal or alteration of the marks. This is typically obtained using multiple copies of the same file, containing different fingerprints. According to [31], by averaging these copies the fingerprint is altered. In general, *Schonberg* and *Kirovski* [32] claim that watermarking is not secure with current technology. However, *Shan He* and *Min Wu* [33] are much more optimistic about the security provided by watermarking techniques. Far from trying to settle the dispute, we simply note that all the current security mechanisms are based on fingerprinting schemes.

Note that to increase the security of the proposed approach, one could use stronger cryptographic tools (e.g, public-key certificates, Internet-based procedure), but since our mechanism is designed for devices with limited computational resources, the usage of such tools is avoided for performance reasons.

[2] For instance, Nokia N16, as well as the iPhone, has 16GB of internal memory.

5 Conclusions

In this paper we presented MOLE, an architecture that produces secure, portable, and customizable video lectures for the mobile scenario. The novelties introduced by MOLE are: i) video lectures are produced in a format that is widely accepted by modern mobile devices; ii) a video lecture contains several lesson paths so as to meet different learning levels requirements; iii) contents are protected against un-authorized usage by using a security mechanism whose computational lightness ensures an easy usage over current mobile devices.

These characteristics allow MOLE to produce video lectures for most mobile devices causing these to be considered as learning tools. As a result, we think MOLE is a candidate approach to ease the process of learning across time and space.

References

[1] K. D. Kelsey (2000), Impact of communication apprehension and communication skills training on interaction in a distance education course, *Journal of Applied Communications*, Vol. 84, No. 4, pp. 7–92.

[2] R. Benlamri, J. Berri, Y. Atif (2006), A framework for ontology-aware instructional design and planning, *Journal of E-Learning and Knowledge Society,* Vol. 2, No. 1, pp. 83–96.

[3] J.R. Corbeil, M.E. Valdes-Corbeil (2007), Are You Ready for Mobile Learing?, *Educase Quarterly*, November 2007, pp. 51–58.

[4] H. S. Keng Siau, F. F.-H. Nah (2006), Use of a classroom response system to enhance classroom interactivity, *IEEE Transaction on Education,* Vol. 49, No. 3, pp. 398–403.

[5] M. C. Wang, G. D. Haertel, H. J. Walberg (1992), What influences learning? A content analysis of review literature, *Journal on Educational Resources,* Vol. 84, No. 1, pp. 30–43.

[6] R. Y.-L. Ting (2005), Mobile learning: Current trend and future challenges, in: *Proceedings of the IEEE International Conference on Advanced Learning Technologies* 2005, IEEE Computer Society.

[7] P. F. Whelan (1997), Remote access to continuing engineering education (RACeE), *IEE Engineering Science And Education Journal,* pp. 205–211.

[8] A. P. Massey, V. Ramesh, V. Khatri (2006), Design, development, and assessment of mobile applications: the case for problem-based learning, *IEEE Transactions on Education,* Vol. 49, No. 2, pp. 183–192.

[9] M. Sharples (2000), The design of personal mobile technologies for lifelong learing, *Computers & Education* Vol. 34, pp. 177–193.

[10] M. Berry, M. Hamilton (2006), Mobile computing, visual diaries, learning and communication: changes to the communicative ecology of design students through mobile computing, in: ACE ’06:

Proceedings of the 8th Austalian conference on Computing education, Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 2006, pp. 35–44.

[11] R. Nachmian (2002), A research framework for the study of a campus-wide web-based academic instruction project, *Internet and Higher Education* Vol. 5, No. 3, pp. 213–229.

[12] J. Traxler (2003), m-learning: Evaluating the effectiveness and cost, in: *Proceedings of the 2nd Annual MLEARN Conference*, 2003, pp. 70–71.

[13] J. T. Black, L.W. Hawkes (2006), A prototype interface for collaborative mobile learning, in: IWCMC '06: *Proceeding of the 2006 international conference on Communications and mobile computing*, ACM Press, New York, NY, USA, 2006, pp. 1277–1282.

[14] Y. Zhang, S. Zhang, S. Vuong, K. Malik (2006), Mobile learning with bluetooth-based e-learning system, in: IWCMC '06: *Proceeding of the 2006 international conference on Communications and mobile computing*, ACM Press, New York, NY, USA, 2006, pp. 951–956.

[15] M. Virvou, E. Alepis (2005), Mobile educational features in authoring tools for personalized tutoring, *Computers & Education,* Vol. 44, pp. 53–68.

[16] L. F. Motiwalla (2007), Mobile learning: A framework and evaluation, *Computers & Education*, Vol. 49, No. 3, pp. 581–596.

[17] Y.-K. Wang (2004), Context awareness and adaptation in mobile learning, in: *Proceedings of the 2nd international workshop on wireless and mobile technologies in education*, IEEE Press, 2004.

[18] A. Syvanen, R. Beale, M. Sharples, M. Ahonen, P. Lonsdale (2005), Supporting pervasive learning environments: Adaptability and context awareness in mobile learning, in: *Proceedings of the 2005 IEEE International Workshop on Wireless and Mobile Technologies in Education*, IEEE Computer Society, 2005.

[19] Y. Cao, T. Tin, R. McGreal, M. Ally, S. Coffey (2006), The athabasca university mobile library project: increasing the boundaries of anytime and anywhere learning for students, in: IWCMC '06: *Proceeding of the 2006 international conference on Communications and mobile computing*, ACMPress, New York, NY, USA, pp. 1289–1294.

[20] D. Marinelli, S. M. Stevens (1998), Synthetic interviews: the art of creating a 'dyad' between humans and machine-based characters, in: *Proceedings of the 4th IEEE Workshop on Interactive Voice Technology for Telecommunications Applications*, 1998.

[21] T. Liu and C. Choudary (2007), Scalable Coding and Wireless Streaming of Lecture Videos for Mobile Learning, *Advanced Technology for Learning*, Vol. 4, No. 2, 2007

[22] H.-Y. Kung, M.-Y. Wu (2005), The design and implementation of an adaptive mobile learning mechanism, in: *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies* (ICALT05), 2005.

[23] E.R. Weippl (2007), Security Considerations in MLearning: Threats and Countermeasures, *Advanced Technology for Learning*, Vol. 4, No. 2.

[24] MPEG4, Overview of the MPEG- 4 Standard, Research report, MPEG Group, [on-line] Available at http://www.chiariglione.org/mpeg/standards/-mpeg-4/mpeg-4.htm (2002).

[25] MPEG7 home page, in: http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm.

[26] S. Cheng, H. Yu, Z. Xiong (2002), Enhanced spread spectrum watermarking of MPEG-2 AAC audio, in: *Proceedings of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Vol. 4, Orlando, FL, USA, 2002, pp. 3728–3731.

[27] RFC 3839: MIME type registrations for 3rd Generation Partnership Project (3GPP) Multimedia files, 2004.

[28] S.Ravi, A. Raghunathan, P. Kocher, S. Hattangady (2004), Security in embedded systems: Design challenges, *ACM Transactions on Embedded Computing Systems*, Vol. 3, No.3, pp. 461–491.

[29] C. Mclvor, M. McLoone, J. V. McCanny (2003), Fast montgomery modular multiplication and RSA cryptographic processor architectures, in: *Proceedings of Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, USA, 2003, pp. 379–384.

[30] K. Biddle, P. England, M. Peinado, B. Willman (2002), The darknet and the future of content distribution, in: *Proceedings of the ACM Workshop on Digital Rights Management*, Washington, DC, USA, 2002.

[31] S. He, M. Hu (2004), Performance study of ECC-Based Collusion-Resistant Multimedia Fingerprinting, in: *Proceedings of the 38th Conferences on Information Sciences and Systems, Princeton*, NJ, USA, 2004, pp. 827–832.

[32] D. Schonberg, D. Kirovski (2004), Fingerprinting and Forensic Analysis Of Multimedia, in: *Proceedings of the 12th annual ACM international conference on Multimedia*, New York, NY, USA, 2004, pp. 788–795.