Journal of Applied Science and Agriculture, 9(20) Special 2014, Pages: 7-12

AENSI Journals

Journal of Applied Science and Agriculture

ISSN 1816-9112

Journal home page: www.aensiweb.com/JASA

Formalization of The Video Structure

Noraida Haji Ali, Fadilah Harun, Noor Maizura Mohamad Noor

School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu, Terengganu, Malaysia

ARTICLE INFO

Article history:
Received 30 September 2014
Received in revised form
17 November 2014
Accepted 25 November 2014
Available online 13 December 2014

Keywords:

ABSTRACT

Background: Formal methods are mathematically based techniques and tools for the formal specification, verification and program validation of software and hardware systems. It able to support program development process. Structuring the video clips is an important part in the video process. A hierarchical structure is a basic structure for the video, it contains four types of component, video, scene, shot, and key frame. Formalizing able to support the integrity and precision of the modeling language. Nowadays, many studies are discovering related to formal methods, video structure, formal specification, and formalizing. This paper discusses about an analysis of formal methods, video hierarchical structure, and formalization of the video basic structure. The output of this study is that developers can easily determine the structure of relationships in the video using the formal specification.

To Cite This Article: Noraida Haji Ali, Fadilah Harun, Noor Maizura Mohamad Noor., Formalization of The Video Structure. ***J. Appl. Sci. & Agric.,*** *9(20): 7-12, 2014*

INTRODUCTION

Formal methods are system design techniques that use strictly specified mathematical models for describing a system. Unlike other design systems, formal methods as mathematical proof as a complement of system testing in order to ensure right behavior (Collins, 1998). Through the use of formal verification scheme, the basic principles of the system must be proved right before they accepted, but it the formal method is different from other design systems (P.Bowen, 1995). The traditional system design has used extensive testing to verify behavior, but testing is capable of only finite conclusions. In contrast, once a theorem is proven true, it remains true. It is very important to note that formal verification does not obviate the need for testing (P.Bowen, 1995). Formal verification cannot fix bad assumptions in the design, but it can help identify errors in reasoning which would otherwise be left unverified. In several cases, engineers have reported finding weakness in systems once they reviewed their designs formally (M.Wing, 1990). Formal design can be seen as a three step process;

1. *Formal Specification:*

In formal specification phase, a system engineer strictly explains a system using a modeling language. Modeling language is a grammar that allows users to model the complex structure of the type specified. It serves a notation, a universe of objects and ccurate rules that define objects that fulfill each relationship.

2. *Formal Verification:*

Formal methods are different from other system specification with emphasis on provability and truth. To build a system that uses a formal specification, designers actually develop a set of theorems of the system. Formal verification is a difficult process, because system has a lot of theorems, each of which must be proved. Formal verification is a complete representation of mathematical notations. A statement verifies through the system descriptions. Proven can be done using the tools of formal methods (Z/Eves).

3. *Program Validation:*

Once the model has been established and validated, it is possible to change the specification to the code. As the difference between software and hardware design grows narrow, formal methods for developing embedded systems have been developed. Validation program is to check the system requirements of the full state earlier.

The Z specification language is a formal specification language used to describe and decomposing specification into a schema to model a computer system (Spivey, 1998). A multimedia database is a structure of multimedia data and it able to handles multimedia information for content retrieval (Jalal, 2001). It supports the

Corresponding Author: Noraida Haji Ali, School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu.
E-mail: aida@umt.edu.my

huge range of multimedia data types like texts, images, graphics, audio and video (Jalal, 2001). Video is the most challenging and has lots of issues than the other type of multimedia data (Affendey,Mamat, 2007). This is because the video consists of a combination of all multimedia data types into a single data flow. This is also caused by the problems had by developers when performing video enhancement one moves from images to image sequences, or video clips and every single video program has their rules and format. Structure is important phase in the design of system development. Our study is to approach a structuring the video using formal specifications methods. Segmentation-based techniques used in structuring a video structure. We hope in this study can enhance the structure of the video system and it can be more efficient and consistent to retrieve their content. This study explains and identifies a formal method as a suggested method to use in the invention and structuring video system.

1. *Related Work:*

Structuring a video used several techniques, such as temporal and spatial (Duda andKeramane, 1995) relations to design the database. However, structuring a video has some constraints, example in relations and modeling the complex object for indexing, searching and retrieval (Ghafoor, 2006). The available structures of the videos are not mutually consistent and not effiecient. These problems leds to determine the specifications of a new structure for video. Earlier researchers have marked the issues related to video and discussed to solve the issues required in producing a general-purpose video system. Structuring a video used several techniques, such as temporal, spatial, and spatiotemporal relations. All the available techniques focused on design an efficient database. Nevertheless, structuring a video has some limitations, example in size and complex object models in different types for indexing, searching, retrieval and organization methods (Ghafoor, 2006). Research before trying to resolve issues related to multimedia databases using temporal specification techniques, such as temporal relationship is to determine the relationship between the duration of the multimedia objects (Harun,Ali, 2013). In a study done by (Djeraba andBriand, 1997), it uses the power of Petri net to model the temporal video and determine the interactive relation. Some other research uses a novel indexing technique based-on. These techniques it is an efficient and effective compression of space to determine its estimated similarity searching in huge multimedia databases (Tuncel,Ferhatosmanoglu, 2002). Structuring the database can be served by a clear video and it can also be specified in the requirement for their content, but the main problems are to find and determine the contents of the video. So far, there is no approach that is used to support video system and view schema objects (Harun,Ali, 2013).

2. *Video Structure:*

Video is the most effective and latest ways for capturing, recording, copying and broadcasting of moving images in the real world around us (Chua andRuan, 1995). It is also the best combination of medium moving objects, photo-realistic images and sounds. Video has become a major source of entertainment over the past three decades. The video also shows the ideal medium for a complex concept and dynamic (change over time) is not easily explained or shown using text or other media. Video database is to provide random access to video data sequentially. Segmentation-based techniques used in a basic video database structure because each individual shots or scenes is a meaningful unit (Jian,Hui, 2006). The hierarchical structure of video material based on hierarchical data model helps users to find and retrieve relevant video materials with efficient way.

Even the others do not have such a structure as the scene content and video news, but a lot of broken frames of video sequences recorded from a single camera movement (shots) that can also explain the structure of video content. The basic structure of the video is a hierarchical structure formed by the video program, scene, shot and key frame. As in Figure 1, video program usually divided into several scenes, and each scene must have one or more shots. A static image and logic unit of video can be supported with a structural system called frames. A shot is an uninterrupted section a series of video frame with static or repeated camera movement. A scene is a series of shots that are ordered from the sequential point of perspective. A video data model based on video segmentation possesses a hierarchical structure, as seen in Figure 1 (Catarci,Donderler, 2003). In Figure 1 show, a video stream partition into several scenes, every scene consist a sequence of shots and every single shot contains a sequence of video frames. A database management system able to support hierarchical video data model.

Informal specification of video:

Different informal representations can be used to explains the same requirement or scenario. Informal representations are a natural language that used to present scenario of the system. While other video content, and scene structure as video news, but a lot of video is a broken frame sequence recorded from a single camera movement (shots) that can also express the structure of video content. The basic structure of the database contains scenes of video, shot and framework. Informal specifications of video can be explained details; firstly video is divided into shots. Secondly, defines the key frames to represent the shots and finally, scenes are built based on the key frames. Table 1 showed functionalities and constraints of operations in the video system. It has three basic operations, add, delete and search record from the system.

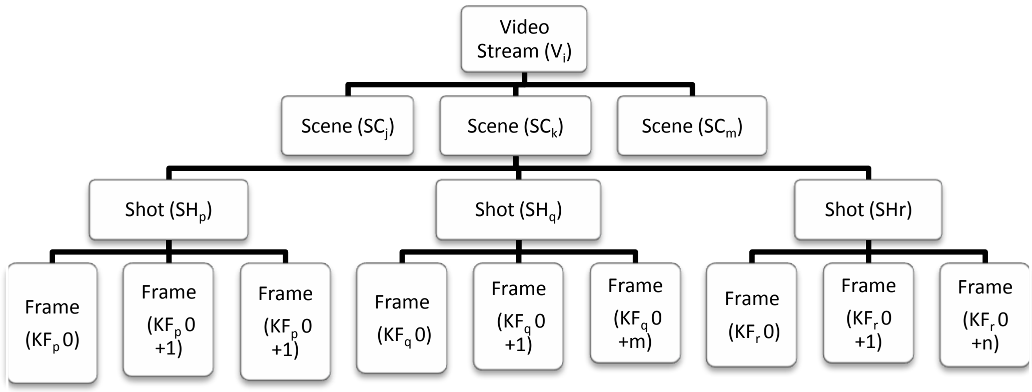

Fig. 1: Hierarchical Structure of Video.

Table 1: Functionalities and Constraints of Operations.

Functionalities	Constraints	Response
Add a record		
Add	The record not in database, response success. Otherwise, The record already in the database, the response fails.	'SUCCESS' 'FAIL'
Remove a record		
Remove	If the record is already in the database, response success. Otherwise, The record not in the database, response not exist.	'SUCCESS' 'NOT EXIST'
Search a record		
Search	If the record is in the database, the response is success. Otherwise, The record not in the database, the response is not exist.	'SUCCESS' 'NOT EXIST'

3. *Formal Specification using Z:*

Specifications and design are connected with each other. It is important for structuring a specification and specification process to have an architecture design (Jusoh,Saman, 2011). The definition of formal specifications is to present a mathematical notation that includes vocabulary, syntax and semantics are formally declared (Jusoh,Saman, 2011). In formal specifications, system requirements and the system design are elaborated in detail, thoroughly analyzed and checked before starting the implementation.

Basic Type and Global Variable:

A structure representation can be described in the following algebraic relations. Declaration of set types is (Ali andHarun, 2014):

[MediaObjects] = sets of media

MediaObjects is a set of media in the video structure, and the variables are;

SC, SH, KF: || MediaObjects

The following free type represents the set of output message to construct all the operations in the video structure.

Response::= *SUCCESS* | *isMediaObjects* | *notMediaObjects* | *FAIL* | *NOTEXIST*

Where:

V: video

SC: sequence of shots.
SH: unbroken sequence of frames.
KF: selected from a shot to represent the contents of the shot.
V (SC, SH, KF)
SH SC
KF SH

State Schema:

The state of the system must contain all information that relates to the video structure. The below is the state schema for the video which consist of Video and set types *MediaObjects*.

Video Þ ⁄ *SC: || MediaObjects; SH: || MediaObjects; KF: || MediaObjects*

⎵ *SC MediaObjects SH MediaObjects KF MediaObjects*╲

Operations Schema:

Now, the operation schema defines the successful cases of operations to add, remove, and search record from the *MediaObjects*.

i. *Add a record:*

Add a record in the video. The new *MediaObjects* will be added to the video if the record does not exist in the video. An above idea is to specify an operation to add a new Scene to the *MediaObjects*. For state schema Add, the inclusion of ⅄*Video* defines that operation potentially changes the state. The potential new Scene will be an input for the operation, declared as;

newSC?: MediaObjects

For this operation, it is important that *newSC* is not already on *MediaObjects.* This leads to the precondition;

newSC? SC

The above properties constitute the preconditions for the *Add* operation. The effect of the operation, captured by the postconditions, is to add the new Scene to the set of *MediaObjects* and if the scene don't exist, the response showed ‘SUCCESS’.

SC' = SC ⋃ *newSC?*
r! = SUCCESS

The required schema to add new scene in the record is;

AddSC Þ ⁄ Δ*Video; newSC?: MediaObjects* ⎵ *newSC? SC SC' = SC* \ *newSC?*╲

ii. *Delete a record:*

This operation removes a record in the video. If the record exists in the *MediaObjects*, so it allowed the process to delete the record. Firstly, the potential of the delete Scene declared as;

dSC?: MediaObjects

For this operation, it is important that *dSC?*is already on *MediaObjects.* This leads to the precondition that checks a scene already exist in the set or not;

dSC? ⊙ *MediaObjects*

If the scene already stored in *MediaObjects*, the response showed ‘SUCCESS’ and the scene can be deleted. Otherwise, if the record doesn't exist, the response showed ‘FAIL’.

SC' = SC \ *dSC?*
r! = SUCCESS

Journal of Applied Science and Agriculture, 9(20) Special 2014, Pages: 7-12

The operation schema to delete a scene in the record is:

DelSC Þ ⁄Δ*Video; dSC?: MediaObjects* ⎵ *dSC?* ⊙ *MediaObjects*   *SC' = SC* \ ⏝*dSC?*⏝⟍

iii. *Search a record:*

This operation is to find records on video. During this operation, *vSC* as an input to view existing record. This operation will not change the state, and therefore includes *Video*. The *vSC* must be a member of the *MediaObjects*.

vSC?: MediaObjects
r!: Response

Firstly, it checks a record already exist in the video or not, and the declaration is;

vSC? ⊙ *MediaObjects*

If the record already exists, the response showed ‘SUCCESS’ and the record can be found. Otherwise, if the record doesn't exist, the response showed ‘FAIL’.

r! = SUCCESS

The schema is:

ViewSC Þ ⁄Δ*Video; vSC?: MediaObjects; r!: Response* ⎵ *vSC?* ⊙ *MediaObjects*   *r! = SUCCESS*⟍

iv. *Error handling schemas:*

The following free type represents the set of output messages required to construct total versions of the above operations. The *SUCCESS* message is used to indicate that an operation has been successfully completed using the following schema:

SuccessResponse Þ ⁄*r!: Response* ⎵ *r! = SUCCESS*⟍

In this section, we are given an example of error schemas handling operation. Based on the view record operation, the precondition exceptions and the schemas to handle as follows:

NotExists Þ ⁄Δ*Video; vSC?: MediaObjects; r!: Response* ⎵ *vSC?*   *MediaObjects*   *r! = NOTEXIST*⟍

The precondition exception occurs when the *vSC?* is not in *MediaObjects*. This is handled by the schema above.

v. *Total operations schema:*

The following free type represents the set of output messages required to construct total versions of the above operations, and for reports from the query operations which are defined above. The situation where a potential new scene is already exists gives the following schema.

TotalOperation Þ ⁄Δ*Video; newSC?: MediaObjects; outcome!: Response* ⎵ *newSC?* ⊙ *SC*   *outcome! = isMediaObjects*⟍

The success message is used to indicate that an operation has been successfully completed, using the following schema in horizontal form;

SuccessAdd Þ ⁄*outcome!: Response* ⎵ *outcome! = SUCCESS*⟍

The complete specification of the *Add* operation can now be defined by combining the various schemas using the propositional operators of the schema calculus encountered above;

TotalAdd Þ *(Add SuccessAdd)* _ *TotalOperation*

4. Discussion:

In segmentation-based techniques, video data model consists of three types of video sequence, namely videos, scenes and shots. Selected key frames of a video sequence content, which presents an image of this subtype class. Video entity can also associate with the text annotation in the video database. However, in video database posses temporal characteristics and it's different with image data. Structuring video streams play a meaningful part in the process of video database. A hierarchical structure is the basic structure for design a video. The process in the simple framework for video hierarchical structure is to separate continuous video frames into discrete physical shots, extract features from video shots and construct a scene structure based on shots (Jian,Hui, 2006). By using formal methods, it can help the process of designing a new structure of video relation algebra. It also can improve the structure of video and can be more efficient to retrieve their content in the structure.

5. Conclusion:

In this study, we explain and elaborate a Z specification of a case study in structure of video system. It explains the relation in the video system between the scene, shot and key frame. This paper focused on early stage in formal specification methods. After the formal specification stage complete, it’s continuing to the formal validation of the theorem. Proving process in the formal specification, it used a long and repetitious a lot of theorems. But, if the proving is done manually without using any prover tools, the possibility of errors can be high. Z/Eves prover tools can help the proving process fast, efficient, and reliable. After this, our future works is to deal with complete and precise specification video database systems and validation of the theorem.

ACKNOWLEDGMENTS

This research was supported by Fundamental Research Grant Scheme (FRGS), Vot: 59298.

REFERENCES

Affendey, L.S., A. Mamat, *et al*., 2007. "Video Data Modelling To Support Hybrid Query." IJCSNS, 7(9): 53.

Ali, N.H. and F. Harun, 2014. Structuring Video Database using a Formal Methods Approach. MMEDIA 2014, The Sixth International Conferences on Advances in Multimedia.

Bowen, P.J., 1995. "Seven More Myths Of Formal Methods." IEEE.

Catarci, T., M.E. Donderler, *et al*. 2003. "BilVideo: a video database management system." MultiMedia, IEEE, 10(1): 66-70.

Chua, T.S. and L.Q. Ruan, 1995. "A video retrieval and sequencing system." ACM Transactions on Information Systems (TOIS), 13(4): 373-407.

Collins, M., 1998. "Formal Methods." Retrieved 6 July, 2014, from http://www.ece.cmu.edu/~koopman/des_s99/formal_methods/.

Djeraba, C. and H. Briand, 1997. Temporal and interactive relations in a multimedia database system. Multimedia Applications, Services and Techniques—ECMAST'97, Springer, 457-473.

Duda, A. and C. Keramane, 1995. Structured temporal Composition of Multimedia Data. iw-mmdbms.

Ghafoor, A., 2006. "Multimedia Support Infrastructure (MSI)." National Science Foundation Retrieved 20 November, 2013, from http://www.cs.purdue.edu/.

Harun, F., N.H. Ali, *et al*. 2013. New Structure of Multimedia Database using Formal Method: Preliminary Study. 12th International Annual Symposium on Sustainability and Management (UMTAS 2010). Kuala Terengganu, Malaysia.

Jalal, S.K., 2001. "Multimedia Database: Content and Structure." Workshop on Multimedia and Internet Technologies.

Jian, F.C., L. Hui, *et al*. 2006. Video Hierarchical Structure Mining. Communications, Circuits and Systems Proceedings, 2006 International Conference on.

Jusoh, J.A., M.Y.M. Saman, *et al*. 2011. "Formal Validation of DNA Database Using Theorem Proving."

Spivey, J.M., 1998. The Z Notation:A Reference Manual, J. M. Spivey Oriel College, Oxford, OX1 4EW, England.

Tuncel, E., H. Ferhatosmanoglu, *et al*. 2002. VQ-index: An index structure for similarity searching in multimedia databases. Proceedings of the tenth ACM international conference on Multimedia, ACM.

Wing, M.J., 1990. "A Specifier's introduction To Formal Method." Defense Advanced Research Project Agency (DOD).